



UNIVERSIDADE ESTADUAL DE CAMPINAS
Faculdade de Engenharia Elétrica e de Computação

RODRIGO HÜBNER

DESENVOLVIMENTO DE UM SISTEMA SSVEP-BCI PARA O
AUXÍLIO EM TOMADA DE DECISÕES

Campinas

2018

RODRIGO HÜBNER

DESENVOLVIMENTO DE UM SISTEMA SSVEP-BCI PARA O
AUXÍLIO EM TOMADA DE DECISÕES

Tese apresentada à Faculdade de Engenharia Elétrica e de Computação da Universidade Estadual de Campinas como parte dos requisitos exigidos para a obtenção do título de Doutor em Engenharia Elétrica, na área de Engenharia de Computação.

Orientador: Prof. Dr. Gilmar Barreto

Coorientadora Profa. Dra. Linnyer Beatrys Ruiz Aylon

ESTE EXEMPLAR CORRESPONDE À VERSÃO
FINAL DA TESE DEFENDIDA PELO ALUNO
RODRIGO HÜBNER, E ORIENTADA PELO PROF.
DR. GILMAR BARRETO

Campinas

2018

Agência(s) de fomento e nº(s) de processo(s): Não se aplica.

ORCID: <https://orcid.org/0000-0001-8454-650>

Ficha catalográfica
Universidade Estadual de Campinas
Biblioteca da Área de Engenharia e Arquitetura
Rose Meire da Silva - CRB 8/5974

H863d Hübner, Rodrigo, 1985-
Desenvolvimento de um sistema SSVEP-BCI para o auxílio em tomada de decisões / Rodrigo Hübner. – Campinas, SP : [s.n.], 2018.

Orientador: Gilmar Barreto.

Coorientador: Linnyer Beatrys Ruiz Aylon.

Tese (doutorado) – Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação.

1. Eletroencefalografia. 2. Interfaces cérebro - Computador. 3. Potenciais evocados. 4. Sistemas de suporte de decisão. 5. Tomada de decisão. I. Barreto, Gilmar, 1958-. II. Aylon, Linnyer Beatrys Ruiz. III. Universidade Estadual de Campinas. Faculdade de Engenharia Elétrica e de Computação. IV. Título.

Informações para Biblioteca Digital

Título em outro idioma: Development of a SSVEP-BCI system for decision-making assistance

Palavras-chave em inglês:

Electroencephalography

Brain-Computer Interfaces

Evoked potentials

Decision support systems

Decision-making

Área de concentração: Engenharia de Computação

Titulação: Doutor em Engenharia Elétrica

Banca examinadora:

Gilmar Barreto [Orientador]

Clenio Figueiredo Salviano

José Tarcísio Franco de Camargo

Rafael Ferrari

Tiago Fernandes Tavares

Data de defesa: 28-09-2018

Programa de Pós-Graduação: Engenharia Elétrica

COMISSÃO JULGADORA - TESE DE DOUTORADO

Candidato: Rodrigo Hübner RA: 180869

Data de Defesa: 28 de setembro de 2018

Título da Tese: “Desenvolvimento de um sistema SSVEP-BCI para o auxílio em tomada de decisões”.

Prof. Dr. Gilmar Barreto (Presidente, FEEC/UNICAMP)

Prof. Dr. Clenio Figueiredo Salviano (CTI Renato Archer)

Prof. Dr. José Tarcísio Franco de Camargo (UNIPINHAL)

Prof. Dr. Rafael Ferrari (FEEC/UNICAMP)

Prof. Dr. Tiago Fernandes Tavares (FEEC/UNICAMP)

A ata de defesa, com as respectivas assinaturas dos membros da Comissão Julgadora, encontra-se no processo de vida acadêmica do aluno.

Agradecimentos

Agradeço primeiramente a Deus por me conceder forças e sabedoria para concluir este trabalho que representa a realização de um sonho profissional.

Agradeço à minha esposa Amanda que me incentivou e ajudou, além do seu amor a apoio incondicional.

Agradeço ao meu orientador Prof. Gilmar, por aceitar a orientação partindo de uma pequena ideia que se transformou, ganhou volume, muito trabalho e se tornou uma tese. Agradeço também por compartilhar sua sabedoria, pela disponibilidade e pela dedicação.

Agradeço minha co-orientadora Prof^a. Linnyer, por sempre acreditar no meu trabalho, mesmo nos momentos que eu não acreditava, pelo incentivo à pesquisa e cooperação de trabalho, pela disponibilidade e pela dedicação.

Agradeço ao Prof. Paulo Oliveira que me ajudou com os primeiros passos da pesquisa de Interfaces Cérebro-Computador, no qual eu nunca tive contato antes do doutorado.

Agradeço a todos os participantes que disponibilizaram o seu tempo para a realização dos experimentos e coleta de dados de EEG.

Agradeço pela disponibilidade e pelo conhecimento técnico passado pelo Prof. Tiago Tavares e Romis Attux.

Agradeço aos colegas e amigos que fizeram parte da turma Dinter UTFPR/FEEC-UNICAMP 2015 (que na verdade tudo começou em fevereiro de 2014), desde àqueles que trabalharam no documento para a proposta do doutorado, em especial o Prof. Roberto Neli, até aos que contribuíram nos estudos dos créditos realizados.

Agradeço a todos os professores da DACOM, em especial ao Prof. Juliano Foleiss pela disponibilidade e conhecimento passado.

Agradeço ao Prof. Heron que não mediu esforços para ajudar na compra de equipamentos utilizados nesta tese, tornando-a possível.

Por fim, agradeço a todos que colaboraram, direta ou indiretamente para a realização deste trabalho.

Agradeço também a UNICAMP, UTFPR, UEM e o CNPq pelo apoio financeiro.

Resumo

Nos últimos anos, Interfaces Cérebro-Computador (BCI) passaram a ter um maior foco em problemas fora do escopo clínico. Sistema BCI podem ser utilizados para controlar equipamentos elétricos e eletrônicos, controle de jogos digitais, etc. A capacidade de poder “controlar” em um sistema BCI, pode ser adaptada a uma ação que auxilia um indivíduo em tomada de decisões, por exemplo, decidir se paramos ou continuamos a conduzir um automóvel ao visualizar os estados de um semáforo de trânsito. O BCI baseado no paradigma de Potenciais Evocados Visualmente em Regime Estacionário (SSVEP), pode ser utilizado para diferenciar alvos com diferentes frequências de cintilação por meio de estímulos visuais. Esta tese de doutorado teve como objetivo avaliar o estímulo SSVEP de altas e baixas frequências admitas pelo paradigma, para a construção de um sistema SSVEP-BCI para auxiliar na tomada de decisões. Para cumprir com este objetivo, foram realizados (1) experimentos com uma base de dados pública com estímulos SSVEP armazenados, para avaliar os códigos desenvolvidos, (2) construção de uma base de dados gerada por meio de experimentos realizados com um protótipo de semáforo de trânsito, para avaliar o funcionamento do protótipo e do equipamento de eletroencefalografia (EEG) e, por fim, (3) experimentos realizados com quatro participantes para avaliar os estímulos SSVEP em baixas frequências de cintilação, tradicionalmente utilizadas do paradigma e altas frequências de cintilação configuradas em um limiar não visível aos nossos olhos, permitindo que o protótipo se comporte de forma mais próxima a situações reais e ainda forneça uma menor fadiga visual. Os resultados obtidos forneceram a exatidão dos programas desenvolvidos para avaliar os estímulos SSVEP e também o funcionamento do protótipo e do equipamento de EEG. Além disso, os experimentos realizados com os quatro participantes apresentaram em média uma acurácia de $89,37\% \pm 8,26\%$ para baixas frequências e $80,62\% \pm 7,18\%$ para altas frequências, no qual concluímos que o sistema SSVEP-BCI pode ser utilizado para auxiliar em situações de tomada de decisão em ambas as faixas de frequência.

Palavras-chaves: EEG; BCI; SSVEP; Tomada de Decisão.

Abstract

In recent years, Brain-Computer Interfaces (BCI) have an increased focus on problems outside the clinical scope. BCI system can be used to control electrical and electronic equipment, control of digital games, etc. The ability to “control” in a BCI system can be adapted to an action that assists an individual in decision-making, for example, deciding whether to stop or continue driving a car when viewing the states of a traffic light. The BCI paradigm based on Stead-State Visually Evoked Potentials (SSVEP) can be used to differentiate targets with different frequencies of scintillation through visual stimuli. This PhD thesis aimed to evaluate the SSVEP stimulus of high and low frequencies admitted by the paradigm, for the construction of a SSVEP-BCI system to assist in decision-making. In order to comply with this objective, we performed (1) experiments with a public database with stored SSVEP stimuli to evaluate developed codes, (2) we constructed a database generated through experiments carried out with a traffic light prototype, to evaluate the functioning of the prototype and electroencephalography (EEG) equipment, and finally (3) experiments was performed with four participants to evaluate the SSVEP stimuli at low scintillation frequencies, traditionally used in the paradigm and high scintillation frequencies configured in a threshold not visible to our eyes, allowing the prototype to behave more closely to real-world situations and still provide less visual fatigue. The results obtained provided the correct execution of written programs to evaluate the SSVEP stimuli and also the functioning of the prototype and the EEG equipment. In addition, the results from the experiments carried out with the four participants presented on average an accuracy of $89.37\% \pm 8.26\%$ for low frequencies and $80.62\% \pm 7.18\%$ for high frequencies, in which we concluded that the SSVEP-BCI system can be used to assist in decision-making situations in both frequency bands.

Keywords: EEG; BCI; SSVEP; Decision-Making.

Lista de ilustrações

Figura 3 – Esquema geral de um sistema SSVEP.	27
Figura 4 – Gráfico de comparação entre um periodograma (preto) e o método Multi Taper aplicado do mesmo sinal.	30
Figura 5 – Gráficos demonstrando a separação dos dados utilizando o classificador SVM.	32
Figura 7 – Configuração geral do experimento. Figuras de (VILIC, 2014).	45
Figura 8 – Fluxo geral de execução dos experimentos apresentando os algoritmos utilizados em cada etapa.	46
Figura 9 – Visão geral do modelo de classificação dos dados ST aplicado na base de dados AVI.	47
Figura 10 – Resultados do experimento com os dados <i>single target</i> da base de dados AVI.	48
Figura 11 – Gráficos da Densidade Espectral de Potência pelo método Multi Taper, da primeira sessão realizada pelo Participante 3 no experimento com os dados <i>single target</i> da base de dados AVI.	49
Figura 12 – Gráficos da Densidade Espectral de Potência pelo método Multi Taper da primeira sessão realizada pelo Participante 4 no experimento com os dados <i>single target</i> da base de dados AVI.	50
Figura 13 – Resultados do experimento com os dados <i>multi target</i> da base de dados AVI.	51
Figura 14 – Gráficos da Densidade Espectral de Potência pelo método Multi Taper da primeira sessão realizada pelo Participante 3 no experimento com os dados <i>multi target</i> da base de dados AVI.	52
Figura 15 – Semáforos construídos com LEDs utilizados no protótipo do experimento 2.	53
Figura 16 – Esquema elétrico de um semáforo com 4 LEDs.	54
Figura 17 – Representação do fluxo do experimento 2.	55
Figura 18 – Ilustrações do protocolo referente ao experimento 2.	56
Figura 19 – Gráficos da Densidade Espectral de Potência de um sinal evocado em 18 Hz com ruído em aproximadamente 16,5 Hz.	57
Figura 20 – Acurácia dos resultados obtidos da validação cruzada do experimento 2.	59
Figura 21 – Gráficos da Densidade Espectral de Potência de frequências evocados em 8 e 10 Hz.	59
Figura 22 – Elementos modificados no protocolo referente ao 3º conjunto experimental.	62

Figura 23 – Fotos da realização do 3º conjunto experimental com um dos participantes.	63
Figura 24 – Divisão dos dados resultante de duas sessões de cada participante. . . .	64
Figura 25 – Divisão dos dados para a otimização do parâmetro C do classificador SVM.	65
Figura 26 – Distribuição das acurácias obtidas na 1ª configuração pelos participantes na janela máxima de tempo de 6 segundos.	67
Figura 27 – Gráficos da Densidade Espectral de Potência de frequências evocados em 10 e 12 Hz.	69
Figura 28 – Distribuição das acurácias obtidas pelos participantes em todas as janelas de tempo para baixas frequências.	70
Figura 29 – Gráficos da Densidade Espectral de Potência de frequências evocados em 8 Hz em um janelamento de 2 segundos.	71
Figura 30 – Distribuição das acurácias obtidas na 2ª configuração pelos participantes na janela máxima de tempo de 6 segundos.	72
Figura 31 – Gráficos da Densidade Espectral de Potência de frequências evocados em 37 e 40 Hz.	73
Figura 32 – Gráficos da Densidade Espectral de Potência de frequência evocada em 43 Hz.	74
Figura 33 – Distribuição das acurácias obtidas pelos participantes em todas as janelas de tempo para altas frequências.	75
Figura 34 – Distribuição das acurácias obtidas pelo participante 3. No gráfico à esquerda está a distribuição de acurácias otimizando os canais somente da janela de 6 segundos. No gráfico à direita está a distribuição de acurácias otimizando os canais em todas as janelas de tempo.	76
Figura 35 – Gráfico de barras para o valor médio de PSD encontrado em cada sessão, divididos por participante, indicando cada cor de LED.	77
Figura 36 – Gráficos da Densidade Espectral de Potência do canal 0z utilizando o método Multi Taper na frequência evocada em 40 Hz.	78
Figura 37 – Fluxograma exemplificando a modelagem de um sistema <i>online</i> SSVEP-BCI para auxiliar na tomada de decisões.	80

Lista de tabelas

Tabela 1	–	Combinações de canais utilizados em CAR e a média obtida por cada participante na janela de tempo de 6 segundos da 1ª configuração . .	68
Tabela 2	–	Significância estatística em valor-p para os janelamentos de tempo de cada participante, tomando como base a janela máxima de 6 segundos	71
Tabela 3	–	Combinações de canais utilizados em CAR e a média obtida por cada participante na janela de tempo de 6 segundos da 2ª configuração . .	73
Tabela 4	–	Significância estatística em valor-p para os janelamentos de tempo de cada participante, tomando como base a janela máxima de 6 segundos	76

Lista de Abreviaturas e Siglas

BCI:	<i>Brain-Computer Interface</i>
CAR:	<i>Common Average Reference</i>
CCA:	<i>Canonical Correlation Analysis</i>
DAFE:	<i>Discriminant Analysis Feature Extraction</i>
ECoG:	Eletrocorticografia
ECG:	Eletrocardiografia
EEG:	Eletroencefalografia
EMG:	Eletromiografia
EOG:	Eletrooculografia
FFT:	<i>Fast Fourier Transform</i>
FIR:	<i>Finite Impulse Response</i>
GC:	<i>Gaussian Classifier</i>
KNNC:	<i>K Nearest Neighbor Classification</i>
LCD:	<i>Liquid Cristal Display</i>
LDA:	<i>Linear Discriminant Analysis</i>
LED:	<i>Light Emitting Diode</i>
MEG:	Magnetoencefalografia
MT:	<i>Multi Target</i>
NBC:	<i>Naive Bayes Classifier</i>
NWFE:	<i>Nonparametric Weighted Feature Extraction</i>
P300:	<i>Peak 300</i>
PCA:	<i>Principal Component Analysis</i>
PSD:	<i>Power Spectral Density</i>
RFE:	<i>Recursive Feature Elimination</i>

RV: Realidade Virtual

SNR: *Signal-to-Noise Ratio*

SS: *Standard Scaler*

SSEP: *Steady-State Evoked Potential*

SSVEP: *Steady State Visually Evoked Potential*

ST: *Single Target*

STFT: *Short-Time Fourier Transform*

SVM: *Support Vector Machine*

TCLE: Termo de Consentimento Livre e Esclarecido

UDP: *User Datagram Protocol*

VEP: *Visually Evoked Potential*

:

Lista de Programas

Programa 1.1	Trecho de código em Python como parte de um arquivo	21
Programa 1.2	Trecho de código em Python como uma seção interativa	21
Programa 4.1	Trecho de código que apresenta a geração do gráfico de PSD pelo método Multi Taper	49
Programa 4.2	Trecho de código que apresenta a geração do gráfico de PSD pelo método Multi Taper	58
Programa 5.1	66

Sumário

1	Introdução	17
1.1	Objetivos	18
1.2	Apresentação da tese	19
1.3	Lista de publicações	20
1.4	Dicionário de termos e notações	20
2	Fundamentação	23
2.1	BCI	23
2.1.1	Aquisição de dados EEG	23
2.2	Paradigmas BCI	25
2.2.1	Atenção Seletiva	26
2.3	Imagética	28
2.4	Processamento de sinais no paradigma SSVEP	28
2.4.1	Pré-processamento de sinais	28
2.4.2	Extração de características	29
2.4.3	Seleção de características	31
2.4.4	Classificação	31
2.5	Auxílio na tomada de decisões em BCI	33
2.6	Considerações finais	35
3	Trabalhos relacionados	36
3.1	Desenvolvimento de um sistema SSVEP-BCI soletrador baseado adotando um teclado de estilo QWERT	36
3.2	Desempenho dos estímulos visuais em movimento baseado em SSVEP-BCI	37
3.3	Um novo método de estimulação SSVEP-BCI de várias classes usando frequências de intermodulação	38
3.4	Direções da Otimização dos Parâmetros de Estímulos para Interfaces Cérebro-Computador Baseadas em Potenciais Visualmente Evocados em Regime Estacionário	38
3.5	Uso de Estímulos Visuais de Alta Frequência Acima da Frequência Crítica de Cintilação em um BCI baseado em SSVEP	39
3.6	Um Método Prático para Detecção de Sonolência na Direção através do Processamento dos Sinais de EEG Estimulados com Luz Cintilante Externa	40
3.7	Classificação única experimental de correlatos neurais de comportamento antecipatório durante a condução real de veículos	40
3.8	Reconhecimento de Épocas P3 de Único Julgamento em um Ambiente Virtual	41
3.9	Classificação das respostas cognitivas do motorista usando análise não-paramétrica de único julgamento	42

3.10	Considerações finais	42
4	Construção e avaliação de sistemas SSVEP-BCI	43
4.1	Base pública SSVEP-BCI	44
4.1.1	Descrição da base pública AVI SSVEP	44
4.1.2	Carregamento e preparação dos dados	45
4.1.3	Procedimentos experimentais	46
4.1.4	Análise dos resultados	47
4.1.5	Considerações finais	51
4.2	Desenvolvimento e avaliação de um sistema SSVEP-BCI com protótipos de semáforos com frequências cintilantes usando LEDs	52
4.2.1	Descrição dos equipamentos utilizados	52
4.2.2	Procedimentos experimentais	54
4.2.3	Análise dos resultados	57
4.2.4	Considerações finais	60
5	Desenvolvimento e avaliação de um sistema SSVEP-BCI para auxiliar em tomada de decisões	61
5.1	Procedimentos experimentais	61
5.2	Análise dos resultados	67
5.2.1	1ª configuração	67
5.2.2	2ª configuração	72
5.2.3	3ª configuração	76
5.3	Discussão sobre a aplicação do sistema SSVEP-BCI em tempo real	78
5.4	Considerações finais	80
6	Conclusões e trabalhos futuros	81
	Referências	84
	APÊNDICE A Programas desenvolvidos	89
A.1	Código da avaliação do 1º experimento com os dados <i>single target</i>	89
A.2	Código da avaliação do 1º experimento <i>multi target</i>	92
A.3	Código Arduino para o desenvolvimento do semáforo com frequência cin- tilante	94
A.4	Código Arduino para encontrar a máxima frequência cintilante não visível ao olho humano	96
A.5	Código para a gravação dos dados de EEG e marcadores no 2º conjunto experimental	98
A.6	Código para a validação e análise dos resultados do 2º conjunto experimental	101
A.7	Código para a validação e análise dos resultados do 3º conjunto experimental	103
A.8	Biblioteca utilizada para geração dos gráficos de PSD	107

A.9	Código para estimar o valor médio de PSD do sinal SSVEP evocado por cores de LED	107
APÊNDICE B	Termo de Consentimento Livre e Esclarecido	109
APÊNDICE C	Parecer do Comitê de Ética	114

1 Introdução

Nas últimas décadas, diversas foram as contribuições para o surgimento de alguns exames médicos que permitem mensurar a atividade cerebral de um indivíduo. Alguns destes exames como é o caso da ressonância magnética (MEG) e a eletroencefalografia (EEG) começaram também a ser utilizados para realizar a interatividade entre o cérebro humano e objetos elétricos e eletrônicos que hoje fazem parte do meio em que vivemos. Tal interatividade deu origem ao estudo que hoje é chamado de Interface Cérebro-Computador, mais conhecido na literatura como BCI (do inglês *Brain Computer-Interface*) (PRASHANT *et al.*, 2015; FOUAD *et al.*, 2015; CHAUDHARY *et al.*, 2016).

Atualmente, o estudo de BCI pode também ser utilizado para o desenvolvimento de sistemas que podem melhorar a qualidade de vida das pessoas que possuem algum tipo de limitação física, seja visual, auditiva ou motora. Além disso, é possível generalizar este conceito para que um sistema BCI possa contribuir na tomada de decisões utilizando estratégias bem concebidas, conhecidas como paradigmas. Entre eles, o paradigma baseado em Potencias Evocados Visualmente em Regime Estacionário ou SSVEP (do inglês *Steady-State Visually Evoked Potential*) (LIN *et al.*, 2014; MARTIŠIUS; DAMAŠEVIČIUS, 2016; LIN *et al.*, 2016; ZHU *et al.*, 2010), é utilizado para determinar um objeto-alvo que um indivíduo está focado, no qual este objeto por sua vez, deve estimular alguma frequência de cintilação (alternar entre cores mais claras e mais escuras e/ou luzes acesas e apagadas em uma determinada frequência). Esta estimulação de frequência, evoca um sinal SSVEP que espera-se estar em torno da mesma frequência que a estimulada. Um equipamento de EEG pode ser utilizado para mensurar este sinal evocado, posicionando os eletrodos sobre o couro cabeludo do indivíduo na região do córtex visual, no qual é a região mais adequada para mensurar este tipo de sinal. Um sistema BCI pode usufruir deste paradigma para a realização de uma decisão entre outras possíveis, por meio de objetos-alvo apresentados em diferentes frequências de cintilação e dada a classificação correta entre o objeto estimulante e o sinal SSVEP evocado, uma ação resultante é encadeada.

O paradigma SSVEP é útil em sistemas BCI para caracterizar uma ação no meio entre outras possíveis. Por exemplo, o sistema pode ser utilizado para o controle de uma cadeira de rodas elétrica, por um indivíduo que não possui suas atividades motoras (MISTRY *et al.*, 2018). Seria possível configurar quatro diferentes alvos piscando em diferentes frequências na tela de um monitor direcionado para o indivíduo, que ao focar em um destes alvos, fará com que o sistema BCI transforme o sinal SSVEP obtido em uma ação que pode ser: (1) ir para frente, (2) ir para trás, (3) ir para a esquerda e (4) ir para a direita. Este tipo de sistema recebe o nome de sistemas SSVEP-BCI.

O conceito do paradigma SSVEP em caracterizar um determinado alvo em uma ação, pode ser transformado em um modelo de tomada de decisões. Neste caso, o sistema BCI pode ser desenvolvido para determinar uma “ação correta” entre outras disponíveis ou incorretas. O alvo neste caso poderia indicar uma tomada de decisão no qual um sistema BCI pode auxiliar nesta escolha indicando ou inferindo na qual é considerada correta. Situações potencialmente de risco necessitam ainda mais de sistemas que podem oferecer este tipo de auxílio, como é o caso da decisão a ser tomada em um semáforo de trânsito ou então ao visualizar a lanterna traseira de um carro sinalizando uma frenagem, enquanto estamos dirigindo. No caso de um semáforo de trânsito, se a luz vermelha estiver acesa, a ação que o motorista consequentemente deverá realizar é frear o carro até pará-lo, e quando o semáforo trocar o seu estado para a luz verde, o motorista continuaria conduzindo o veículo normalmente, com exceção da luz amarela que nos alerta a reduzir a velocidade do automóvel e “prestar atenção” próximo a um cruzamento. O desenvolvimento de um sistema BCI que auxilie em casos como este, poderia contribuir diretamente na prevenção de acidentes.

1.1 Objetivos

Meios de transporte, manipulação de máquinas industriais e outros exemplos ou situações que geram constantemente a necessidade de tomada de decisões, estão cada dia mais “autônomos”, porém eventuais falhas nestes sistemas, devolvem o controle da máquina ao ser humano, no qual muitas vezes não possui instruções necessárias para lidar com a situação recorrente, originando assim erros, acidentes e até mesmo catástrofes. Esta tese de doutorado teve como objetivo viabilizar um sistema baseado em Interfaces Cérebro-Computador, para auxiliar uma pessoa a tomar a decisão correta principalmente em situações consideradas de risco.

Tomando como princípio este objetivo, foi desenvolvido e avaliado um sistema SSVEP-BCI no formato *offline* (em tempo não real), focando no sinal SSVEP em baixas e altas frequências de cintilação. Isto foi necessário para viabilizar um sistema que pudesse auxiliar na tomada de decisões. Alguns trabalhos já demonstraram a possibilidade de criar sistemas SSVEP-BCI em altas frequências de cintilação (SAKURADA *et al.*, 2015; CHABUDA *et al.*, 2018), buscando condições para maximizar os estímulos de sinais SSVEP. A experimentação *offline* realizada nesta tese, permitiu que fosse avaliado condições menos propícias para a ocorrência da evocação do sinal SSVEP tanto nas baixas quanto nas altas frequências de cintilação, buscando em tais condições de estimulação, um cenário mais próximo de situações reais. Desta forma, o foco do trabalho foi concentrado nos métodos de processamento do sinal SSVEP em tal condição, para viabilizar um sistema que auxilie na tomada de decisões em situações reais. Além disso, o objetivo de trabalhar com as altas frequências de cintilação (acima de 30 Hz) foi de diminuir a fadiga visual (MOLINA

et al., 2009), evitar problemas relacionados a convulsões epiléticas induzidas pela baixa frequência de cintilação (HARDING *et al.*, 2005) e fazer com que o estímulo fique mais próximo de situações reais, no qual a frequência de cintilação não seja visível aos nossos olhos.

Para cumprir com este objetivo, primeiramente foram reproduzidos um conjunto de experimentos baseados no paradigma SSVEP utilizando uma base de dados pública, com a intenção de avaliar os códigos desenvolvidos. Prosseguindo com os experimentos, bases de dados foram construídas por meio de aquisições de sinais EEG, utilizando o protótipo de semáforo de trânsito construído com Diodos Emissores de Luz (do inglês, *Light Emitting Diode* - LED) para simular uma condição de tomada de decisão em uma situação real. Por fim, quatro participantes saudáveis foram convidados para realizar experimentos com o sistema configurado em baixas e altas frequências de estimulação SSVEP, por meio do protótipo desenvolvido, para que pudessemos realizar uma avaliação detalhada e viabilizasse o sistema SSVEP-BCI para auxiliar na tomada de decisões. Os passos específicos para alcançar o objetivo, foram:

1. Desenvolver programas para reproduzir experimentos SSVEP utilizando bases de dados públicas;
2. Desenvolver um protótipo de semáforo de trânsito utilizado para estimular o sinal SSVEP em diferentes frequências de cintilação;
3. Realizar experimentos com baixas e altas frequências de cintilação com quatro participantes para a obtenção de uma base de dados;
4. Avaliar de forma mais detalhada a base de dados obtida pelos participantes, para viabilizar a construção de um sistema SSVEP-BCI para auxiliar na tomada de decisões.

1.2 Apresentação da tese

A redação desta tese é dividida em seis capítulos, incluindo este capítulo de introdução. Os outros capítulos estão apresentadas como segue. O Capítulo 2 apresenta a fundamentação teórica relacionada aos itens estudados. O Capítulo 3 apresenta os trabalhos relacionados que contribuíram para o sistema desenvolvido e implementado nesta tese. O Capítulo 4 apresenta os componentes utilizados para o desenvolvimento e análise de sistemas SSVEP-BCI, utilizando uma base pública e um protótipo construído para estimular o sinal SSVEP. O Capítulo 5 apresenta a construção e avaliação *offline* de um sistema SSVEP-BCI para avaliar tomada de decisões em semáforos de trânsito, utilizando como alvo, LEDs com altas e baixas frequência de cintilação. O Capítulo 6 apresenta as conclusões obtidas desta tese,

além da discussão sobre trabalhos e perspectivas futuras. Por fim, é apresentada a lista de referências bibliográficas utilizadas e nos Apêndices estão os programas desenvolvidos necessários para a reprodução deste trabalho e os Anexos com a documentação relacionada ao Comitê de Ética e Cidadania.

1.3 Lista de publicações

1. HÜBNER, R.; SANTIAGO, A. N.; MIGUEL, P. V. O.; BARRETO, G. Mechanisms Based on Neural Interface to Alert and Decision Support in Automotive Driving In: Brazilian Institute of Neuroscience and Neurotechnology Congress, 2015, Campinas. Journal of Epilepsy and Clinical Neurophysiology. 2015. v.21.
2. HÜBNER, R.; SILVA, W. M.; MIGUEL, P. V. O.; RUIZ, L. B.; BARRETO, G. Desenvolvimento de Cenários de Interface Cérebro-Computador Baseados em Estímulos Visuais P300 In: XLIV Congresso Brasileiro de Educação em Engenharia, 2016, Natal.
3. HÜBNER, R.; SANTIAGO, A. N.; MIGUEL, P. V. O.; RUIZ, L. B.; BARRETO, G. A model based in brain-computer interface to identify risk situations and support to decision-making In: Neuroscience 2016, 2016, San Diego. Society for Neuroscience. , 2016. p.1599 - 1600.
4. HÜBNER, R.; MIGUEL, P. V. O.; RUIZ, L. B.; BARRETO, G. In: Brazilian Institute of Neuroscience and Neurotechnology Congress, 2017, Campinas. Journal of Epilepsy and Clinical Neurophysiology. A SSVEP-BCI model to make correct decisions on traffic signs, 2017. v.23.
5. HÜBNER, R.; RUIZ, L. B.; BARRETO, G. Towards in SSVEP-BCI systems for assistance in decision-making. In: Future Technologies Conference (FTC), 2018, Vancouver, BC, Canada.
6. HÜBNER, R.; FOLEISS, J. H. RUIZ, L. B.; BARRETO, G. A Brain-Computer Interface Based on SSVEP that Uses a High-frequency Stimulus for Decision-making. In: Computational Intelligence and Neuroscience, Hindawi Limited, 2018. (**Em processo de análise pelo corpo editorial**).

1.4 Dicionário de termos e notações

É destacado o significado de algumas palavras ou expressões utilizadas durante todo o texto como segue:

- Análise/experimento *offline*: quando os dados obtidos durante a experimentação serão analisados **após** a experimentação prática com o(s) participante(s);
- Análise/experimento *online*: quando os dados obtidos durante a experimentação serão analisados **durante** a experimentação prática com o(s) participante(s);
- Cintilação: entende-se nesta tese que este termo refere-se a oscilação de um objeto-alvo piscando (alternando entre um contraste mais claro e mais escuro ou ligado e desligado). Será frequente o termo aparecer em conjunto com “frequência”, no qual “frequência de cintilação” poderá ser um objeto alvo (no caso um LED) alternando em ligado e desligado, em uma determinada frequência;
- *Feedback*: entende-se por ser uma resposta informativa apresentada ao usuário, após o processamento de dados já obtidos por meio de um sistema BCI.

Durante o texto, serão apresentados alguns códigos, sendo a maioria escritos na linguagem de programação Python. Exemplos de código na linguagem de programação Python podem estar de duas formas diferentes. No formato de um arquivo ou parte dele, conforme o Programa 1.1:

Programa 1.1 Trecho de código em Python como parte de um arquivo

```
1 def fib(n):
2     a, b = 0, 1
3     while a < n:
4         print(a, end=' ')
5         a, b = b, a+b
6     print()
7 print(fib(1000))
```

Ou então no formato de seções interativas em Python, como apresentado no Programa 1.2:

Programa 1.2 Trecho de código em Python como uma seção interativa

```
1 \begin{minted}[frame=lines,mathescape]{Python}
2 >>> numeros = [2, 4, 6, 8]
3 >>> produto = 1
4 >>> for numero in numeros:
5 ...     produto = produto * numero
6 ...
7 >>> print('O produto é:', produto)
8 O produto é: 384
```

Em seções interativas, o marcador “>>>” que inicia qualquer construção da linguagem. Se a construção produz uma saída, isto é mostrado na linha seguinte sem nenhum marcador.

Linhas iniciadas com `...` são parte de um bloco de código que está abaixo do marcador que o iniciou por `>>>`.

2 Fundamentação

Este capítulo trata da fundamentação teórica de BCI, tendo como foco o paradigma de Atenção Seletiva SSVEP bem como as estratégias utilizadas no processamento de sinais deste paradigma. Além disso, será apresentada uma discussão sobre o auxílio à tomada de decisões por meio de um sistema BCI.

2.1 BCI

Uma BCI possui para a maior parte das aplicações, um esquema genérico especificado pelos seguintes passos: (1) Aquisição de dados cerebrais; (2) Pré-processamento dos sinais; (3) Extração e seleção de características, (4) classificação; e, por fim (5) Aplicação e *feedback*. Um esquema tradicional de um sistema BCI conforme mencionado, é apresentado pela Figura 1.

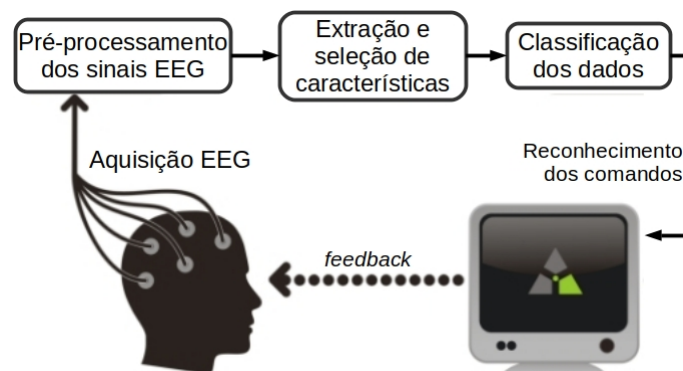


Figura 1 – Esquema tradicional de um sistema BCI. Adaptado de: OPEN-i¹.

2.1.1 Aquisição de dados EEG

É possível captar bio-sinais por meio de transdutores em quase toda extremidade do nosso corpo. Estes sinais representam atividades orgânicas ou um comportamento específico desencadeado por uma série ações fisiológicas (THAKOR, 1999). Os métodos para medir a atividade cerebral podem ser realizados por meio de cirurgia (invasiva) e sem a realização dela (não-invasiva).

Para a realização da aquisição dos sinais pelo método invasivo, uma cirurgia é necessária para abrir a caixa craniana do paciente e implantar eletrodos diretamente no córtex cerebral, método este chamado de eletrocorticograma (ECoG). Métodos invasivos combinam

¹ Fonte: <https://openi.nlm.nih.gov/detailedresult.php?img=PMC3112189_pone.0020674.g001&req=4>

um sinal mensurado de boa qualidade, uma melhor resolução espacial e uma alta faixa de frequência. Neste método, há uma grande redução nos artefatos que afetam o sinal mensurado. Porém, motivos éticos, financeiros e outros, inviabilizam em muitos casos o uso desta técnica, abrindo exceções apenas para casos em que o paciente necessita de um sistema BCI para se comunicar ou realizar outras tarefas fundamentais.

O método não-invasivo também pode ser utilizado para mensurar bio-sinais encontrados em todo o corpo humano, por meio de técnicas como o eletrocardiograma (ECG), o eletromiograma (EMG), o eletroencefalograma (EEG) ou o eletro-oculograma (EOG) (BRONZINO; PETERSON, 2015). O método não-invasivo, como o nome sugere, não necessita cirurgia e o sinal pode ser mensurado por meio do contato com a pele do indivíduo. O foco desta tese é o método não-invasivo utilizando EEG.

No método de EEG, são utilizados eletrodos para obter um sinal elétrico em contato com o couro cabeludo do indivíduo. Um dispositivo EEG registra sinais elétricos do cérebro chamados de sinais pós-sinápticos do neurônios. As medições de um EEG representam a soma de toda atividade síncrona dos neurônios abaixo da área medida por um eletrodo (WEBSTER, 2009). Um equipamento para aquisição de sinal EEG pode ter um ou até mais de cem eletrodos em contato com o couro cabeludo. A aplicação de um gel ou pasta condutiva pode ser necessário entre o eletrodo e o couro cabeludo, porém em alguns equipamentos EEG, não é necessário utilizar o gel condutivo (eletrodo seco).

Alguns padrões foram desenvolvidos para denotar regiões específicas do posicionamento do eletrodo na região do couro cabeludo. Um dos padrões mais utilizados recebeu o nome de *International 10-20 System* (em português, Sistema Internacional 10-20) (NIEDERMEYER; SILVA, 2004), que também foi utilizado nesta tese, pois a localização dos eletrodos neste padrão é suficiente para o paradigma SSVEP utilizado. O termo 10-20 significa que os principais eletrodos são posicionados em uma proporção de 10, 20, 20, 20, 20 e 10% da região que vai de *Nasion* a *Inion*, somando 21 possíveis eletrodos que podem ser posicionados. O mapeamento dos eletrodos no Sistema Internacional 10-20 é apresentado pela Figura 2. Os rótulos das posições dos eletrodos são relativos aos canais específicos mensurados, onde as primeiras letras denotam a região específica como segue: Fp - pré-frontal, F - frontal, C - central, P - parietal, O - occipital e T - temporal. Há também outros Sistemas Internacionais padronizados em relação às posições dos eletrodos, tais como o 10-10 e 10-5 (JURCAK *et al.*, 2007), no qual possibilita a colocação de 81 e 320 eletrodos respectivamente.

Algumas características podem ser extraídas de um EEG, que podem ser úteis para um exame clínico ou para a utilização em um sistema BCI específico. Uma delas é os seis ritmos cerebrais, distinguidos de acordo com as seguintes faixas de frequência: δ (1-4 Hz), θ (4-7 Hz), α (8-12 Hz), μ (8-13 Hz), β (12-30 Hz) e γ (25-100 Hz). Os ritmos δ e θ

² Fonte: <<https://www.diytdcs.com/tag/1020-positioning/>>

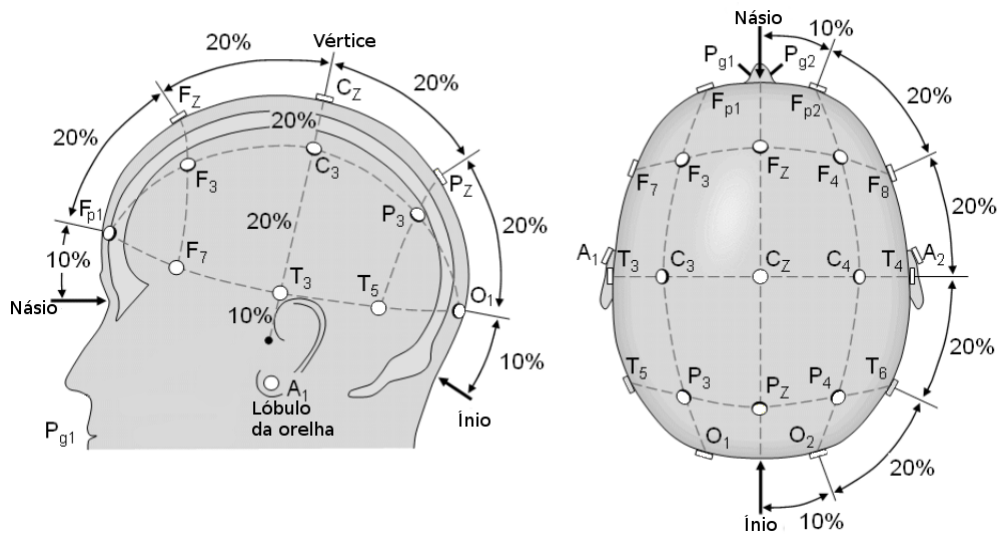


Figura 2 – Sistema Internacional 10-20 para notação de mapeamento de eletrodos. Adaptado de DIY tDCS².

ocorrem em condições de alto estado emocional ou em um estado de sono. O ritmo α ocorre quando o indivíduo está acordado, porém com os olhos fechados e em estado de relaxamento. O padrão do ritmo β ocorre quando está acordado com os olhos abertos. O ritmo γ é adquirido por meio dos sinais obtidos da região do córtex somatossensorial e o ritmo μ da região do córtex sensorimotor (SANEI; CHAMBERS, 2007).

Para a construção de um sistema BCI é necessário o entendimento sobre as regiões do cérebro responsáveis por cada atividade do corpo humano e o posicionamento dos eletrodos para mensurar a atividade cerebral correta. Porém, uma BCI não é capaz de ler a mente ou decifrar o pensamento do indivíduo (GRAIMANN *et al.*, 2010), como é mostrado em alguns filmes de ficção científica, por exemplo em uma cena de Jornada nas Estrelas em que o personagem Spock veste um capacete em outro personagem para “decifrar os seus pensamentos”. Assim, é possível somente detectar padrões específicos de atividades por meio do sinal cerebral associado, utilizando os chamados Paradigmas BCI.

2.2 Paradigmas BCI

Os paradigmas BCI determinam o que e como o indivíduo deve se comportar para produzir determinados padrões conhecidos que podem ser interpretados por um sistema computacional. O indivíduo deve ser submetido geralmente a uma calibração do equipamento e um treino antes da experimentação. A configuração do ambiente físico, posicionamento dos eletrodos e do conjunto de software estão diretamente associados ao paradigma utilizado. Os paradigmas atualmente utilizados são: Atenção Seletiva e Imagética (Mühl *et al.*, 2010).

2.2.1 Atenção Seletiva

Paradigmas BCI baseados em atenção seletiva requerem estímulos externos que resultam em padrões específicos em resposta pelo cérebro (GRAIMANN *et al.*, 2010). Tais estímulos podem ser visuais, auditivos ou táteis. Neste método, cada estímulo é associado a um comando específico e o usuário deverá focar sua atenção em um estímulo alvo para gerar a ação correspondente. Nesta tese, serão utilizados estímulos visuais e os principais paradigmas que utilizam estes estímulos são: Potencial Evocado em Regime Estacionário (do inglês *Steady-State Evoked Potentials* - SSEP) e o P300.

- **SSEP:** Estímulos externos periódicos podem ser evocados no sinal obtido de toda região do córtex visual. Tais estímulos podem ser do tipo sensorial, auditivo, mas principalmente visual, conhecido na literatura como SSVEP.
- **SSVEP:** Os estímulos SSVEP podem ser desencadeados por meio de uma frequência visual estimulada no indivíduo. Normalmente estes estímulos são gerados por uma simulação computacional na tela do monitor, mas também é comum a utilização de LEDs para a geração deste tipo de estimulação (ZHU *et al.*, 2010). Ao utilizar a tela de um monitor, existe a desvantagem da necessidade de configurar o experimento de modo que a frequência de atualização da tela seja múltiplo das frequências de cintilação utilizadas nos objetos alvo, porém, ocorre uma menor quantidade de interferências dos equipamentos físicos utilizados, por conta do isolamento elétrico do próprio monitor. Já ao utilizar LEDs, interferências eletromagnéticas externas podem ocorrer, porém existe uma maior flexibilização para adotar as frequências de estimulação nos alvos. Um alvo por exemplo, pode ser uma luz cintilando a uma frequência de 8 Hz no qual o indivíduo está focado visualmente e desta forma, será possível reconhecer uma resposta no sinal EEG evocado no córtex visual em uma frequência próxima à do estímulo. Em um estudo realizado por Regan (1977), verificou-se que frequências estimuladas podem variar de 5 a 100 Hz, porém a faixa de frequência mais utilizada está entre 5 e 30 Hz, por apresentar uma resolução do sinal SSVEP melhor.

Um esquema geral de um sistema SSVEP é apresentado na Figura 3, no qual uma pessoa com o equipamento de EEG está focada em um alvo com uma determinada frequência de cintilação (A). Então o sinal EEG obtido (B) é transformado computacionalmente no domínio da frequência (C), sendo possível visualizar como resultado a frequência evocada. É possível notar por meio da Figura 3, que a frequência no qual a pessoa está focado no alvo é relativa a f_1 e que múltiplos dessa frequência podem ser visualizados ($2f_1$, $3f_1$, ...), caracterizando o fenômeno da sobreposição harmônica da frequência (CHUMERIN *et al.*, 2013), no qual o projeto de um sistema SSVEP que deve ser levado em consideração no projeto do sistema. O sinal SSVEP

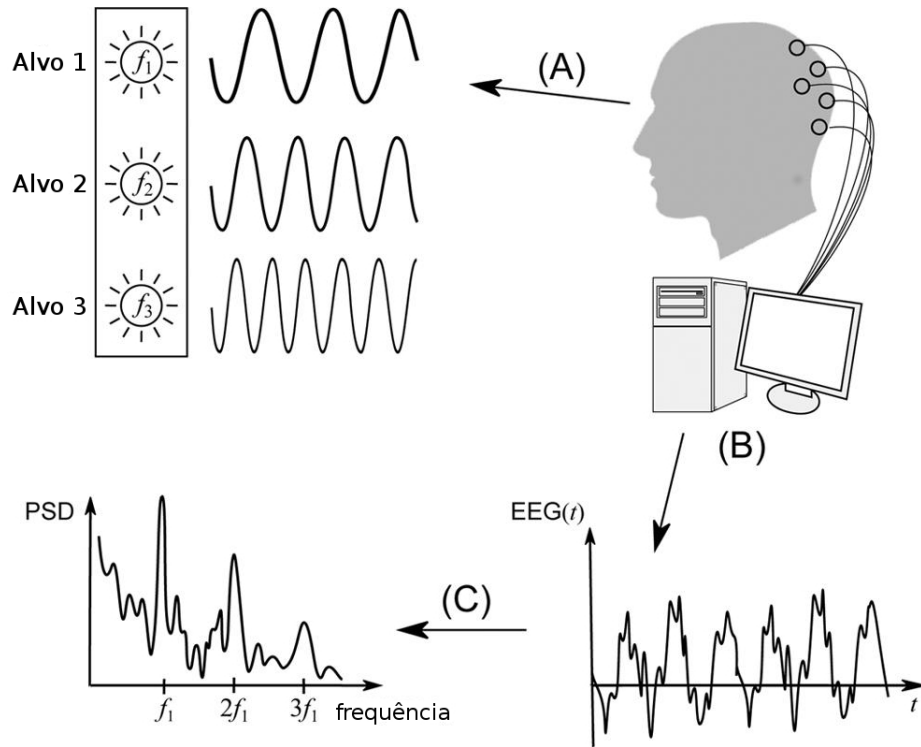


Figura 3 – Esquema geral de um sistema SSVEP.

dispõe de outras características como luminância, contraste e cromática, que podem ser moduladas junto com as frequências de cintilação de um estímulo alvo (CHEN *et al.*, 2017). Na Seção 2.4 são apresentadas com mais detalhes os passos do processamento de sinais SSVEP e como esse paradigma pode ser utilizado para desenvolver uma nova estratégia para o reconhecimento das luzes verde/amarelo/vermelho em um semáforo de trânsito.

- **P300:** O paradigma P300 consiste em obter uma série de picos positivos no sinal EEG mensurado, no qual ocorre uma variação na amplitude do sinal no domínio do tempo durante um pequeno espaço de tempo. Esta variação deve ocorrer após o aparecimento do estímulo alvo infrequente entre vários frequentes (FAZEL-REZAI; AHMAD, 2011). Os estímulos podem ser do tipo auditivo, visual ou sensorial. Um exemplo de estímulo visual pode ser determinado por uma letra ou símbolo da tela de um computador que o indivíduo está focado, que ao receber um contraste (geralmente mais claro), irá gerar um pico no sinal evocado aproximadamente 300 milissegundos após a ocorrência do estímulo. Para este pico se dá o nome de P300 (*Peak 300*). No trabalho de Halder *et al.* (2015) é apresentada uma aplicação geralmente utilizada com o paradigma P300 chamada de soletrador (do inglês, *speller*).

2.3 Imagética

Aplicações em BCI para reabilitação de pacientes que perderam algum movimento motor, também contam com o paradigma de Imagética. A imaginação do indivíduo pode criar padrões que já estão muito bem definidos na literatura. Esta imaginação pode ocorrer de duas formas: motora e não motora.

- **Motora:** A Imagética Motora pode ser vista como um ensaio mental de um movimento do corpo sem que o movimento realmente aconteça (PFURTSCHELLER; NEUPER, 2001). Tais movimentos resultam em uma ativação do córtex motor que, por sua vez, promove oscilações que podem ser obtidas pelo EEG. Essas oscilações são manipuladas por um sistema BCI e são transformadas em ações para uma aplicação específica. Essa técnica geralmente exige muitas sessões de treinamento antes de obter resultados aceitáveis, o que pode se tornar uma tarefa difícil, pois muitas pessoas possuem dificuldade em imaginar movimentos motores (NEUPER *et al.*, 2005).
- **Não motora:** Às vezes imaginar outros objetos, sons e até cheiros podem ser mais fáceis do que imaginar o movimento de partes do corpo. Um paradigma utilizando a Imagética Não Motora é útil para classificar padrões vindos da imaginação de outras coisas que não seja o movimento motor. No trabalho de Faller *et al.* (2014) é realizado um estudo sobre o aprimoramento de sistemas BCI utilizando imagética não motora em pacientes com comprometimento motor severo.

2.4 Processamento de sinais no paradigma SSVEP

Um experimento BCI baseado no paradigma SSVEP está relacionado à forma como são gerados os estímulos para indivíduo e como sinais obtidos por meio do equipamento de EEG são processados. A seguir são apresentados os passos de processamento do sinal SSVEP.

2.4.1 Pré-processamento de sinais

O pré-processamento é um estágio fundamental do tratamento de um sinal EEG, pois o objetivo é remover sinais ruidosos ou indesejáveis, sem perder as informações relevantes. A comparação do nível de um sinal desejado com o nível de ruído existente, é chamado de **relação sinal-ruído** (do inglês *signal-to-noise ratio* - SNR). Quando a SNR for baixa no sinal, significa que padrões detectáveis serão difíceis de serem encontrados. Já quando a SNR for alta no sinal, significa que os padrões serão fáceis de serem identificados. Técnicas de filtragem de sinal são aplicadas nesta etapa de processamento, que ao utilizar uma forma combinada, facilita a determinação dos sinais de interesse.

Filtragens temporais e espaciais são utilizadas como pré-processamento do sinal. Nesta tese foram utilizadas as técnicas de filtragem por passagem de banda (*band pass*) FIR pelo método de janelamento de Hamming na faixa de 5 a 50 Hz e o filtro *notch* em 60 Hz (SHENOI, 2005). Além disso, foi utilizado um método de filtragem espacial chamada de Referência Média Comum (do inglês *Common Average Reference* - CAR) (MCFARLAND *et al.*, 1997). Uma filtragem espacial pelo método CAR subtrai os artefatos comuns a todos os eletrodos. Comumente em BCI, para a redução de sinais não desejados mensurados pelo EEG, são posicionados eletrodos de referência em locais que tentam subtrair os potenciais comuns em todos os outros eletrodos. Um eletrodo de referência pode ser posicionado na região central do couro cabeludo (Cz), processos mastoides e lóbulos da orelha. A técnica de filtragem espacial CAR reduz as componentes que estão em maior número nos eletrodos, subtraindo cada amostra do sinal pelo valor médio da diferença de potencial calculada para todos os eletrodos e a diferença aferida em cada eletrodo. A técnica CAR pode ser definida como:

$$V_i^{CAR} = V_i^{ER} - \frac{1}{N_e} \sum_{j=1}^{N_e} V_j^{ER} \quad (2.1)$$

onde V_i^{ER} é o potencial do i -ésimo eletrodo e N_e é o número de eletrodos utilizados.

2.4.2 Extração de características

Este passo descreve as propriedades esperadas do sinal de entrada na forma de características. Tais características podem ser obtidas utilizando: a forma de onda do sinal analisado no domínio do tempo; Componentes de frequências individuais no domínio da frequência; Espectro de densidade de potência; Análise de frequência de tempo (utilizando por exemplo uma Transformada de Fourier Curta, do inglês Short-Time Fourier Transform - STFT); e não obstante, Modelos Autorregressivos, etc. (YANG *et al.*, 2007). O paradigma SSVEP normalmente utiliza características obtidas por meio de componentes espectrais.

Em sistemas SSVEP-BCI costuma-se utilizar métodos para extração de características baseados nas informações espectrais apresentadas no sinal obtido, pois dado um determinado conjunto de frequências evocadas, o cálculo da densidade espectral de potência (do inglês Power Spectral Density - PSD) pode extrair do sinal as informações de interesse para serem classificadas. Os principais métodos utilizados para a análise de frequência em SSVEP são: Banco de Filtros, Espectrograma, Método de Welch (CARVALHO *et al.*, 2015) e método Multi Taper (MCCOY *et al.*, 1998).

A extração de características tem como resultado um vetor que é utilizado como entrada para as próximas etapas do processamento que compõem o sistema. Desta forma, a extração de características pode ser vista como uma etapa fundamental na preparação do sinal (seleção de características) que posteriormente é classificado e transformado em ações.

O método de Multi Taper foi utilizado como método para a extração de características nesta tese. Este método pode ser aplicado pela ferramenta **MNE-Python**³, utilizada no processamento de sinal SSVEP. O método Multi Taper cobre algumas limitações da análise convencional de Fourier ao extrair informações espectrais de um sinal. A transformada de Fourier é aplicada em uma única janela de amostras de um sinal para fornecer as propriedades espectrais. Desta forma, o PSD obtido acaba por ser uma estimativa tendenciosa do verdadeiro conteúdo espectral. Estes problemas podem ser contornados obtendo a média de vários eventos ao longo do sinal, porém este método não é confiável em um conjunto de dados pequenos. Ao invés disso, o método Multi Taper obtém múltiplas estimativas independentes da mesma amostra. Cada conicidade (*taper*) de dados do sinal é multiplicada por um elemento-base para fornecer uma janela que estima a potência de cada componente de frequência. Cada conicidade resultante são ortogonais em pares, fornecendo estimativas independentes do espectro subjacente e o espectro final é obtido por meio da média de todos os espectros afilados. A comparação entre um periodograma (em preto) e o método Multi Taper aplicado sobre este sinal (em vermelho) está representada na Figura 4.

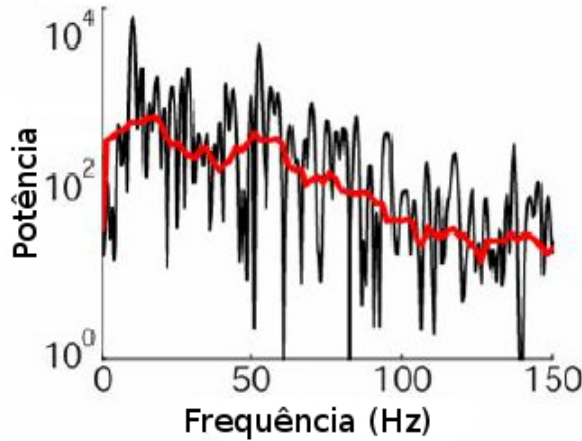


Figura 4 – Gráfico de comparação entre um periodograma (preto) e o método Multi Taper aplicado do mesmo sinal.

O método Multi Taper tenta reduzir a variação das estimativas espectrais usando um pequeno conjunto de conicidades à série temporal. São empregadas as primeiras K conicidades onde $K < 2NW - 1$, no qual N é a quantidade total de conicidades e W é a largura da série, de modo que todas as conicidades tenham um bom confinamento de energia. O mais simples estimador espectral Multi Taper é então definido como:

$$\hat{S}_{mt}^{(N)}(f) = \frac{1}{K} \sum_{k=1}^K \hat{S}_k^{(N)}(f) \quad (2.2)$$

No qual S é entendido em sua literatura como “conicidades Slepian”.

³ <http://martinos.org/mne>

2.4.3 Seleção de características

Um grande número de variáveis que serão analisadas futuramente por um classificador, pode ser obtido na extração de características. Este passo consiste em selecionar as características mais relevantes do conjunto obtido pela extração de características. Utilizando este passo, é possível melhorar o desempenho do classificador em termos de execução mais rápida e eficaz. Entre as técnicas de seleção de características são citadas a de Filtro (Coeficientes de Correlação de Pearson e Índice de Davies-Bouldin) e a técnica de *Wrappers* (CARVALHO *et al.*, 2015). A técnica chamada Eliminação Recursiva de Características (do inglês *Recursive Feature Elimination* - RFE) baseada em *Wrappers* é utilizada nesta tese por apresentar no geral uma melhor performance no mesmo trabalho citado e pela disponibilidade do algoritmo RFE implementado no módulo *Scikit Learn* (HWANG *et al.*, 2018), também amplamente utilizado nesta tese.

O objetivo do RFE é selecionar características por recursividade, considerando os menores conjuntos de características. Primeiramente o estimador é treinado com um conjunto inicial de características e a importância de cada uma delas é obtida por meio de um coeficiente que mede sua importância (OU *et al.*, 2017).

2.4.4 Classificação

A classificação é a fase final de processamento do sinal SSVEP, na qual é possível decidir qual ação ou comando deverá ser executado. A seleção de características tem como saída um vetor de características utilizado pela classificação dos dados em diferentes classes. Os classificadores que seguem a abordagem de aprendizagem supervisionada utilizam amostras de exemplos rotuladas, denominadas conjunto de treinamento. Este conjunto é formado por diversas amostras rotuladas de cada classe, para que o classificador seja capaz de reconhecer novas amostras e classificá-las em alguma das classes que compõem este conjunto.

Existem diversos algoritmos de classificação supervisionados, tais como o de Máquina de Vetores de Suporte (do inglês *Support Vector Machine* - SVM) e Análise Discriminante Linear (do inglês *Linear Discriminant Analysis* - LDA) (ZHANG *et al.*, 2017). Também possuem alguns classificadores conhecidos por terem uma aplicação “mais rápida” na literatura, que é o caso do K-Vizinhos mais Próximos (do inglês *K-Nearest Neighbor* - KNN) (MARACINE *et al.*, 2017). Porém, nesta tese optou-se em utilizar o classificador SVM, com base em seu desempenho apresentado no estudo de Martišius e Damaševičius (2016).

Os SVMs são um conjunto de estruturas de filtragens úteis para classificação, regressão e detecção de dados discrepantes. As principais vantagens em utilizar classificadores SVM é a efetividade em espaços dimensionais elevados, eficácia nos casos em que o número de

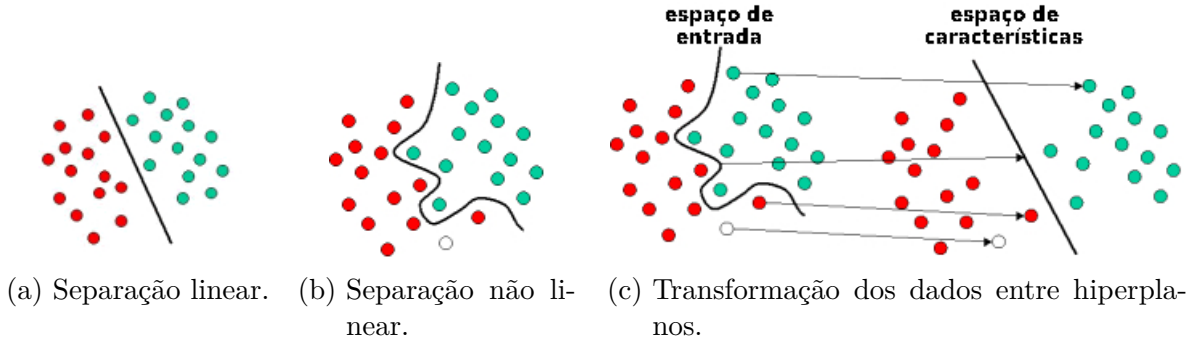


Figura 5 – Gráficos demonstrando a separação dos dados utilizando o classificador SVM.

dimensões é maior que o número de amostras e eficácia em termos de memória utilizada pela utilização dos chamados “vetores de suporte” (SMOLA; SCHÖLKOPF, 2004).

Os vetores de suporte são utilizados na etapa de treinamento para encontrar um plano que melhor define os limites de decisão para a separação dos dados em diferentes classes. Um exemplo desta separação é apresentado na Figura 5a, no qual um classificador linear é utilizado. Neste exemplo, uma linha separa todos os pontos verdes dos vermelhos, porém a maioria das tarefas de classificação exigem estruturas mais complexas para fazer uma separação ideal com base nos exemplos disponíveis para o treinamento, situação que é apresentada pela Figura 5b. Uma boa separação é alcançada pelo hiperplano que possui a maior distância até o ponto dos dados de treinamento mais próximo de qualquer classe, definido como margem funcional. Quanto maior a margem, menor o erro de generalização do classificador. Considerando que o problema da classificação pode ser declarado em um espaço dimensional finito, frequentemente ocorre que as classes não podem ser separadas naquele espaço. Por esta razão, o espaço dimensional finito original é mapeado em um espaço dimensional maior até que presumivelmente seja possível a separação dos dados. Este mapeamento é apresentado na Figura 5c, no qual um conjunto de funções matemáticas conhecida como **kernels** é utilizada para reorganizar os dados em um hiperplano em que estão “visivelmente” separados.

Para construir um hiperplano ideal, o SVM emprega um algoritmo de treinamento iterativo, que é usado para minimizar uma função de erro. Um forma comum de treinamento da SVM define a seguinte função de erro:

$$\frac{1}{2}w^T w + C \sum_{i=1}^N \zeta_i \quad (2.3)$$

sujeito às seguintes restrições:

$$y_i(w^T \phi(x_i) + b) \geq 1 - \zeta_i \quad (2.4)$$

com $\zeta_i \geq 0, i = 1, \dots, N$

Onde C é chamada de constante de capacidade, w é o vetor de coeficientes, b é uma constante e ζ_i representa os parâmetros para manipular os dados não separáveis de entrada. O índice i identifica os N casos de treinamento. Note que $y \in \pm 1$ representa os rótulos de duas classes e x_i representa as variáveis independentes. O *kernel* ϕ é usado para transformar os dados de entrada para o espaço de características. Deve-se notar que quanto maior o C , mais o erro é penalizado. Assim, C deve ser escolhido com cuidado para evitar “sobre ajustes”, que é quando o modelo de treinamento está minimamente generalizado para a classificação dos dados.

O classificador SVM pode utilizar diferentes tipos de **kernel**. Estes incluem *linear*, *polinomial*, *função de base radial (RBF)* e *sigmóide*:

$$K(X_i, X_j) = \begin{cases} X_i \cdot X_j & \text{linear} \\ (\gamma X_i \cdot X_j + C)^d & \text{polinomial} \\ \exp(-\gamma |X_i - X_j|^2) & \text{RBF} \\ \tanh(\gamma X_i \cdot X_j + C) & \text{sigmóide} \end{cases} \quad (2.5)$$

onde $K(X_i \cdot X_j) = \phi(X_i) \cdot \phi(X_j)$

Isto é, a função *kernel* representa um produto escalar de pontos de dados de entrada mapeados para o espaço de características dimensional superior por uma transformação γ . Nesta equação, γ (*gamma*) é um parâmetro ajustável de certas funções *kernel*.

Quando o valor de **gamma** é muito alto, o raio da área de influência dos vetores de suporte inclui apenas o próprio vetor de suporte e nenhuma quantidade de regularização com C será capaz de evitar o sobre ajuste. Quando o valor de **gamma** é muito baixo, o modelo fica muito restrito e não pode capturar a complexidade ou a “forma dos dados”. A região de influência de qualquer vetor de suporte selecionado incluiria todo o conjunto de treinamento. O modelo resultante se comportará de maneira semelhante a um modelo linear com um conjunto de hiperplanos que separam os centros de alta densidade de qualquer par de duas classes. A influência de diferentes valores para os parâmetros **gamma** e C na classificação de um conjunto de dados com duas classes distintas, são mostradas Figura 6.

2.5 Auxílio na tomada de decisões em BCI

O conceito de tomada de decisões na psicologia é considerado como o processo cognitivo que resulta na seleção de uma crença ou da realização de uma ação entre várias possibilidades alternativas. Todo processo de tomada de decisão produz uma escolha final, que pode ou não solicitar uma ação. Desta forma, a tomada de decisão é o processo de

⁴ Fonte: <http://scikit-learn.org/stable/auto_examples/svm/plot_rbf_parameters.html>

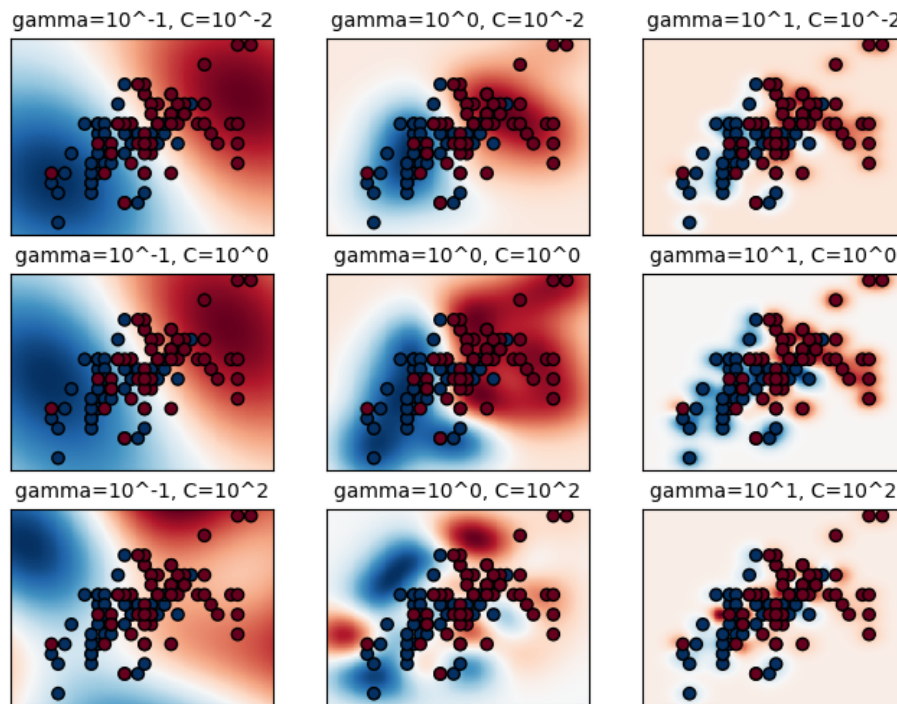


Figura 6 – Gráficos que apresentam a influência de diferentes valores para os parâmetros γ e C na classificação de um conjunto de dados com duas classes distintas. Fonte: Scikit-learn.org⁴.

identificar e escolher alternativas com base nos valores, preferências e crenças do tomador de decisão.

O auxílio à tomada de decisões se caracteriza em contribuir na escolha da melhor decisão imposta a um indivíduo. Tal situação pode ser banal como escolher uma bebida em uma máquina de refrigerante, no qual o gosto pode ser o fator decisivo, ou então uma situação de risco em que se deve decidir em parar ou desviar com um veículo diante de um obstáculo enquanto está dirigindo, no qual a decisão incorreta pode resultar em um acidente.

Nesta tese, o processo decisório é aplicado em semáforos de trânsito, no qual as ações são comumente conhecidas: ao estar ligado na cor verde, o condutor do veículo poderá continuar andando normalmente, na cor vermelha o condutor deverá desacelerar até parar o veículo e, para alguns tipos de semáforos de trânsito, na cor amarela o condutor deverá ter mais atenção no cruzamento, consequentemente reduzindo a velocidade do veículo.

O processo decisório está relacionado com a percepção que o indivíduo possui em uma determinada situação. Segundo Chiavenato em Chiavenato (2014), é ressaltado que qualquer decisão é composta por (1) um tomador de decisão, (2) um objetivo, (3) uma preferência, (4) uma estratégia, (5) uma situação e (6) um resultado. Em nosso sistema BCI em construção, cada componente da tomada de decisão está relacionada respectivamente com (1) o participante do experimento, (2) reconhecer qual luz está ativa por meio do sistema BCI, (3) nula, pois o sistema já está preparado para agir em cada situação do semáforo,

(4) a estratégia de reconhecimento da BCI, (5) simular uma luz corrente no semáforo e, por fim, (6) enviar algum *feedback* ao participante, que em nosso sistema poderá ser sonoro para não perturbar o experimento visual.

2.6 Considerações finais

Este capítulo apresentou a fundamentação dos assuntos trabalhados nesta tese. Além disso, introduziu diversos conceitos fundamentais para o entendimento do leitor durante o restante do texto. O próximo capítulo apresenta alguns trabalhos relacionados que contribuíram para o desenvolvimento desta tese de doutorado.

3 Trabalhos relacionados

Trabalhos realizados em BCI são cada vez mais aplicados em tarefas da rotina das pessoas. Tais tarefas podem incluir: um sistema soletrador para pessoas com limitações na fala; um controle de próteses; um controle de uma cadeira de rodas (SCHUH *et al.*, 2014); o controle parcial ou total de um jogo digital (MARSHALL *et al.*, 2013); controle e interação com dispositivos elétricos e eletrônicos (PERAKAKIS; POTAMIANOS, 2013); e, não obstante o auxílio na tomada de decisões, que é foco deste trabalho, no qual foi utilizado inicialmente o processo decisório com o uso do paradigma SSVEP e posteriormente realizado um estudo mais aprofundado com o mesmo paradigma. Em estudos recentes, sistemas SSVEP-BCI são amplamente utilizados para o reconhecimento de alvos visuais apresentados em uma determinada frequência de cintilação, no qual um indivíduo está focado visualmente.

Este capítulo apresenta trabalhos que utilizaram sistemas baseados no paradigma SSVEP e outros sistemas BCIs, que contribuem para os métodos propostos nesta tese. A revisão foi realizada com base nos seguintes tópicos de pesquisa:

- Sistemas utilizando o modelo SSVEP-BCI tradicional com múltiplos alvos simultâneos;
- Avaliação e comparação de sistemas BCI em sua maioria utilizando SSVEP;
- Discussão e apresentação de novas estratégias no processamento de sinais em sistemas SSVEP-BCI e por fim;
- Trabalhos que não utilizam necessariamente o paradigma SSVEP para detectar ou antecipar situações que necessitam de atenção no trânsito.

Alguns artigos apresentados, também refletem o estado da arte em questões relacionadas ao paradigma SSVEP, quando se trata por exemplo das altas frequências utilizadas em uma aplicação real, o qual é uma contribuição científica firmada por esta tese.

3.1 Desenvolvimento de um sistema SSVEP-BCI soletrador baseado adotando um teclado de estilo QWERT

No trabalho de Hwang *et al.* (2012) originalmente intitulado em inglês “*Development of an SSVEP-based BCI Spelling System Adopting a QWERTY-style LED Keyboard*”, foi introduzido um novo sistema soletrador no modelo de teclados de computador QWERTY,

utilizando 30 LEDs representando cada tecla, cintilando em diferentes frequências. Este método possibilita que o indivíduo possa selecionar um alvo (caractere) sem a necessidade de múltiplos passos como em sistemas de soletradores tradicionais BCI. Os autores conseguiram obter ampla resolução de frequências, reconhecendo estritamente por exemplo um estímulo cintilante com a diferença de 1 Hz entre estímulos consecutivos. Os experimentos foram realizados com dez participantes saudáveis, no qual cinco participaram de um experimento *offline* piloto e os outros cinco em um experimento *online*. Foram utilizadas 68 palavras da língua inglesa para as avaliações e os participantes ficaram a uma distância de 50 cm do teclado de LEDs. Foi observado que os participantes levaram em média 6,38 segundos para reconhecer cada caractere. Para os resultados preliminares *offline*, foram obtidas acurácias de 76,67% e 72,33% para ângulos de visualização 40 e 30 graus respectivamente. Os resultados *online* foram melhores, pois o melhor ângulo e a melhor combinação de eletrodos foram utilizadas (Oz e O2 no sistema 10-20), obtendo acurácias relativas a quantidade de tempo que os participantes levaram para reconhecer cada caractere: 5 segundos: 84,69%, 6 seg.: 86,17% e 7 seg.: 89,53%. Deste trabalho foi possível obter informações importantes sobre a distância e ângulo de posicionamento dos LEDs para um melhor resultado, além dos melhores posicionamentos de eletrodos para este fim.

3.2 Desempenho dos estímulos visuais em movimento baseado em SSVEP-BCI

Sobre desempenho em sistemas BCI, o trabalho de Zhang *et al.* (2016) originalmente intitulado em inglês “*Performance of Virtual Stimulus Motion Based on the SSVEP-BCI*”, abordou um modelo de estimulação em que os objetos alvo estão se movendo constantemente na tela de um monitor. Desta forma, não foi realizado o reconhecimento dos objetos no sinal SSVEP por meio da cintilação do alvo. Invés disso, foi utilizado um método de “escorregamento” baseado nas imagens em movimento na tela. Nos experimentos foram utilizados 5 participantes saudáveis e 6 eletrodos posicionados na região do córtex visual. No experimento, a tela apresentava 4 círculos desenhados movendo-se em diferentes direções. A premissa utilizada para o protocolo SSVEP é que não era necessário cintilar os objetos, pois a atualização da imagem para gerar o movimento já realizava esta tarefa. Contudo, a cintilação no movimento de cada círculo foi ajustada a 10, 11, 12 e 13 Hz. Cada participante realizava 9 seções com diferentes movimentos, repetindo 3 vezes cada uma delas. Cada fase de cintilação era de 5 segundos e o tempo de transição de uma fase para outra de 3 segundos. Uma pausa de 20 segundos foi dada no intervalo de cada seção. Foi utilizada a Análise de Correlação Canônica (do inglês, *Canonical Correlation Analysis* - CCA) para saber para qual objeto o indivíduo estava olhando no experimento *online*, além do método de densidade espectral. Foi obtido como resultado uma acurácia média

de 94,3%. Este trabalho apresenta a possibilidade de explorar o movimento dos objetos alvo como um método de cintilação SSVEP.

3.3 Um novo método de estimulação SSVEP-BCI de várias classes usando frequências de intermodulação

No trabalho de Chen *et al.* (2017) originalmente intitulado em inglês “*A Novel Stimulation Method for Multi-class SSVEP-BCI using Intermodulation Frequencies*”, foi desenvolvido um método utilizando frequências de intermodulação para SSVEP-BCIs com alvos cintilando na mesma frequência de 15 Hz, mas com diferentes frequências de intermodulação, tornando possível codificar uma maior quantidade de alvos. Na metodologia aplicada, os autores codificaram 9 objetos alvo em uma tela de LCD, no qual retângulos foram dispostas em uma matriz 3x3. A frequência de modulação para cada alvo foi gerada por meio das características de cor (“C”, alternando os quadros em verde), vermelho e cinza, característica de luminância (“L”, alternando quadros com diferença de 20 cd/m⁻²) e a mistura das duas (CL) formando três abordagens avaliadas. Em relação aos resultados, a média da acurácia para a avaliação *online* das três abordagens foi de 85%, sendo a mistura das duas (CL) a maior obtida com o resultado de 96,41%. Este trabalho tem como importância apresentar uma alternativa no processamento de sinais SSVEP tradicionais ao possibilitar a classificação de diferentes alvos em uma mesma frequência, porém com a nova abordagem de intermodulação.

3.4 Direções da Otimização dos Parâmetros de Estímulos para Interfaces Cérebro-Computador Baseadas em Potenciais Visualmente Evocados em Regime Estacionário

No trabalho de Duszyk *et al.* (2014) originalmente intitulado em inglês “*Towards an Optimization of Stimulus Parameters for Brain-Computer Interfaces Based on Steady State Visual Evoked Potentials*”, é apresentado uma visão sobre as influências de diversas características do estímulo visual SSVEP na magnitude das frequências obtidas como respostas. Cinco características foram avaliadas para os alvos estimulantes SSVEP: tamanho, distância, cor, forma e presença de um ponto de fixação no meio do cintilante. A distância entre os alvos de estimulação e a presença ou ausência do ponto de fixação não teve efeito significativo nos resultados. Já a cor e o tamanho do alvo cintilante desempenharam um papel importante na resposta SSVEP. Experimentos foram realizados com 5 adultos saudáveis e quatro estímulos foram apresentados na tela do monitor com diferentes frequências de cintilação. Foi adicionado um grupo de LEDs adjacentes a cada objeto mostrado na tela,

responsável por gerar aleatoriamente a luminância imposta. Dois experimentos foram realizados: o primeiro para escolher os melhores parâmetros e o segundo para realizar um estudo mais aprofundado. As respostas espectrais são maiores para a cor branca, seguida de amarelo, vermelho, verde e por fim azul que é a mais baixa. Os parâmetros investigados, tais como cor, tamanho, etc, foram controlados por software e gerados aleatoriamente. Em relação ao tamanho dos objetos apresentados, a qualidade de informação espectral é proporcionalmente maior em relação ao tamanho do objeto. Outras características não surtiram efeitos relevantes para este estudo.

Este trabalho apresentou informações importantes para a caracterização do ambiente no qual é inserido o protótipo de nosso trabalho, além de conter informações importantes de como processar o sinal EEG para cada característica apresentada. Além disso, os autores apresentam uma fundamentação fisiológica da visão, onde explicam que existem 3 principais vias de informação paralelas para o processamento da visão. Isso foi importante para definir quais características podem ser melhor aproveitadas na estimulação visual. Indicam que as cores azul, vermelho e amarelo apresentam diferentes magnitudes na mesma frequência de evocação. Mostram também que o tamanho e a distância em que o alvo é apresentado é crucial para o desenvolvimento de simulações em interfaces gráficas para a resposta SSVEP.

3.5 Uso de Estímulos Visuais de Alta Frequência Acima da Frequência Crítica de Cintilação em um BCI baseado em SSVEP

O trabalho de Sakurada *et al.* (2015) originalmente intitulado em inglês “*Use of High-frequency Visual Stimuli Above the Critical Flicker Frequency in a SSVEP-based BMI*”, tem como objetivo apresentar uma avaliação utilizando frequências acima das tradicionalmente utilizadas em sistemas SSVEP-BCI. O experimento foi conduzido utilizando LEDs na cor verde (com baixa luminância) e azul (com alta luminância). Os autores verificaram como resultado a acurácia do sistema e o nível de fadiga visual dos participantes. Os participantes foram convidados a focarem em uma frequência de cintilação verde e azul (30-70 Hz), e a amplitude (potência espectral) SSVEP foi avaliada. Os participantes foram solicitados a indicar se o estímulo estava visivelmente cintilando e a relatar seu nível subjetivo de desconforto. Também foram avaliadas frequências visíveis (41, 43 e 45 Hz) contra invisíveis (61, 63 e 65 Hz). Como resultado, foi obtido em média de acurácia de 93,1% e 88% para os estímulos visíveis e invisíveis respectivamente. Além disso, concluíram que mesmo aumentando as frequências de cintilação para valores invisíveis, o sistema SSVEP-BCI continua oferecendo uma boa performance e ainda, a fadiga visual foi reduzida nos participantes.

Nesta tese avaliamos as altas frequências de cintilação (invisíveis ao olho humano) para

nos aproximarmos de uma situação real (no caso um semáforo de trânsito real que não possui frequências de cintilação). Desta forma, o trabalho apresentado foi importante para indicar a viabilidade desta estratégia, além de tomarmos conhecimento de que altas frequências também diminuem a fadiga visual dos participantes.

3.6 Um Método Prático para Detecção de Sonolência na Direção através do Processamento dos Sinais de EEG Estimulados com Luz Cintilante Externa

O trabalho de Resalat e Saba (2015) originalmente intitulado em inglês “*A Practical Method for Driver Sleepiness Detection by Processing the EEG Signals Stimulated with External Flickering Light*”, apresenta um detector de sonolência para motoristas. O objetivo é determinar a atenção do motorista, detectando a sonolência através de um sistema BCI. Para a construção de tal sistema foi utilizado o paradigma SSVEP, onde a partir de 4 LEDs dispostos a alguns centímetros do usuário, 2 diretamente a frente e dois em um ângulo de 45 graus a direita e a esquerda, conseguem detectar a atenção do usuário. Tais experimentos utilizaram dois tipos de características, baseadas em *Fast Fourier Transform (FFT)* e *Power Spectral Density (PSD)*. Os classificadores utilizados foram LDA, SVM e Max. Os testes foram realizados com cinco indivíduos, sendo que no melhor caso foi obtido uma acurácia de 98,2%. Nesta tese avaliamos o sistema proposto com quatro participantes, o que nos dá uma boa referência de como o sistema se comporta com um estímulo visual próximo de situações reais.

3.7 Classificação única experimental de correlatos neurais de comportamento antecipatório durante a condução real de veículos

Sobre percepção, cognição e tomada de decisões no trânsito, o trabalho de Khaliliardali *et al.* (2016) originalmente intitulado em inglês “*Single Trial Classification of Neural Correlates of Anticipatory Behavior During Real Car Driving*”, realiza uma discussão sobre a importância de antecipar a cognição de eventos no trânsito, que tem como exemplo saber quando é necessário frear o automóvel, perceber as trocas de luzes acesas ao se aproximar de um semáforo, etc. Apresentam ainda que muitos sensores já concebidos, acoplados em um automóvel podem auxiliar na tomada de decisão e afirmam que a possibilidade de tomar uma decisão correta pode ser maior se utilizarmos um sistema BCI para prever a intenção do condutor. Foi utilizado como método de correlação neural por meio da detecção de padrões, para que pudesse criar um sistema BCI baseado em comportamento antecipatório. O experimento foi realizado com 8 motoristas em diferentes dias em uma

estrada fechada com 6 semáforos em pontos específicos. Os motoristas foram instruídos a pisar no acelerador no sinal verde e pisar no freio no sinal vermelho. Foi desenvolvida uma avaliação de único julgamento observando picos filtrados nas ondas *gamma* e confirmou-se a existência de Potenciais Corticais Lentos antecipados em resposta aos semáforos, que demonstram a possibilidade de detectar esses potenciais na condução em mundo real. Este estudo adiciona a possibilidade futura de não só detectar a cor ativa em um semáforo, mas também a intenção comportamental do condutor do veículo, o que poderia ser proposto como trabalho futuro a esta tese.

3.8 Reconhecimento de Épocas P3 de Único Julgamento em um Ambiente Virtual

O trabalho de Bayliss e Ballard (2000) originalmente intitulado em inglês “*Single Trial P3 Epoch Recognition in a Virtual Environment*”, é o mais antigo de todos citados até o momento, porém possui uma real importância para esta tese quando se trata do reconhecimento de semáforos utilizando uma BCI. Os autores realizaram um experimento em um ambiente de Realidade Virtual (RV), no qual o indivíduo emerge em um cenário em que deve decidir por parar ou não dependendo da luz ativa no semáforo a sua frente. Foi utilizado o paradigma P300, que pelos autores foi chamado de P3. A metodologia baseou-se em manter a luz do semáforo apagada antes do motorista chegar a 30 metros do semáforo. Nos exatos 30 metros, o semáforo ativa a luz laranja para atrair a atenção do motorista e logo após, aleatoriamente, ativa a luz verde (prosseguir) ou vermelha (parar). O sistema de reconhecimento utilizando o P300 foi utilizado para identificar se o motorista irá parar se a luz vermelha estiver ativa no semáforo, logo após o aparecimento da luz laranja. Os experimentos foram realizados com 6 indivíduos e os autores obtiveram uma acurácia média de 84,5% para testes *offline* e 83% para testes *online*. O baixo desempenho foi ainda explicado pela quantidade de artefatos gerados quando o indivíduo é imposto a um ambiente de RV.

Nesta tese, a tomada de decisão leva em consideração quais das luzes no semáforo estão acesas, para viabilizar a possibilidade de auxiliar no que fazer em cada uma das três situações possíveis (verde: prosseguir; laranja: diminuir a velocidade ou interromper a aceleração; vermelho: parar). Não estamos levando em consideração que, em uma situação real, o indivíduo necessita focar no semáforo antes que possa identificar qual luz está acesa, o que já nos sugere um possível trabalho futuro.

3.9 Classificação das respostas cognitivas do motorista usando análise não-paramétrica de único julgamento

No último trabalho a ser apresentado de Lin *et al.* (2007) originalmente intitulado em inglês “*Classification of driver’s cognitive responses using nonparametric single-trial eeg analysis*”, os autores também reforçam a importância da direção segura, sendo importante criar técnicas que ajudam na cognição das tarefas do condutor no trânsito. Foi utilizado o paradigma P300 e o principal objetivo do estudo é comparar três diferentes métodos de extração de características (*nonparametric weighted feature extraction* NWFE, *principal component analysis* PCA e *discriminant analysis feature extraction* DAFE) e três diferentes métodos de classificação (*Gaussian classifier* GC, *k nearest neighbor classification* KNNC e *naive Bayes classifier* NBC), em uma experimentação de julgamento único em um ambiente de RV contendo semáforos. Três participantes foram convidados a realizar a experimentação, no qual foram instruídos a apertar um botão com a mão direita para desacelerar e parar o carro quando visualizar o semáforo vermelho, apertar um botão com a mão esquerda quando visualizar o semáforo em amarelo e não realizar nenhuma ação quando visualizar o semáforo verde. Os autores citam o trabalho apresentado anteriormente, e esclarecem que o seu estudo é estendê-lo para avaliar as diferentes técnicas de extração de características e classificação. Os resultados apresentados demonstraram que o extrator de características NWFE em conjunto com o classificador GC obtiveram as melhores acurácias, sendo em média de 10 a 24% maior que a combinação dos algoritmos DAFE-NBC. Este trabalho apresenta uma combinação de algoritmos que pode ser útil nesta tese, quando são utilizadas frequências de cintilação não visíveis.

3.10 Considerações finais

Nos trabalhos relacionados apresentados foi possível levantar diversas características úteis para o desenvolvimento desta tese, desde a aquisição de dados até a classificação. Foi possível também descobrir outros fatores importantes a serem utilizados na metodologia, tais como: o tempo de seção realizado pelos participantes, o tempo de descanso, quantidade de vezes que cada experimento foi realizado, possíveis cenários de experimentação, posição dos eletrodos para aquisição EEG, etc. No próximo capítulo descreveremos os componentes para avaliação de um sistema SSVEP-BCI, detalhando o desenvolvimento do protótipo, o equipamento de EEG utilizado e os programas desenvolvidos, além de apresentar testes realizados com o sinal SSVEP em baixas frequências de cintilação.

4 Construção e avaliação de sistemas SSVEP-BCI

Este capítulo apresenta a base metodológica para esta tese, no qual buscou-se avaliar um sistema proposto de forma totalmente experimental, gerando resultados qualitativos e quantitativos. As experimentações neste capítulo apresenta os componentes físicos e lógicos necessários para a construção e avaliação de sistemas SSVEP-BCI. Para demonstrar tais componentes, foram desenvolvidos dois conjuntos experimentais que são a base da construção do sistema proposto nesta tese de doutorado. Os dois conjuntos estão divididos da seguinte forma:

1. Desenvolvimento de códigos para a avaliação de uma base de dados pública SSVEP-BCI; e
2. Construção de um protótipo de semáforos de trânsito utilizando LEDs, utilizados como alvos com evocação de frequências cintilantes e adaptação do código (item 1) utilizada para a concepção do sistema SSVEP-BCI proposto.

Os dois conjuntos experimentais possuem uma importância **incremental** para alcançar o objetivo principal desta tese. Inicialmente demonstramos que programas desenvolvidos por nós são consistentes e suficientes para avaliar uma base de dados pública SSVEP. Após isso, um segundo conjunto experimental foi realizado para avaliar uma base de dados gerada no âmbito desta tese, utilizando um protótipo de semáforo de trânsito construído com LEDs, no qual os LEDs realizam estímulos SSVEP tradicionais (frequências baixas, entre 8 e 15 Hz), baseados em alvos cintilantes. Dado a eficácia em reproduzir a experimentação de sistemas SSVEP-BCI nos moldes tradicionais, foi possível projetar um novo sistema BCI para a realização de tomada de decisões, com alvos cintilantes em baixas e altas frequências (não visíveis ao participante), utilizando os mesmos componentes físicos do segundo conjunto experimental. O desenvolvimento do novo sistema e a análise dos resultados está exposto no Capítulo 5.

Para o desenvolvimento de código dos experimentos foram utilizadas as seguintes ferramentas:

- O projeto **MNE-Python** (GRAMFORT *et al.*, 2013), que unifica um conjunto de bibliotecas escritas na linguagem de programação Python com a finalidade de analisar dados de EEG e MEG;

- A biblioteca **Scikit Learn**¹ para rotinas baseadas em Inteligência Computacional, também escrita na linguagem de programação Python; e
- A biblioteca **SciPy**² que unifica diversas funções científicas disponíveis em Python, tais como análises estatísticas, construção de gráficos, etc.

Basicamente, a utilização destas três ferramentas em conjunto, possibilitou trabalhar com todas as etapas dos experimentos de programação, desde a aquisição e pré-processamento até a classificação dos dados.

4.1 Base pública SSVEP-BCI

Nesta seção é apresentado o experimento realizado com a base de dados AVI SSVEP³, desenvolvida por Vilic (2014). Esta base de dados foi construída como parte de um trabalho do mesmo autor (VILIC *et al.*, 2013), no qual foi desenvolvido um “soletrador com suporte a dicionários”. Primeiramente a Seção 4.1.1 apresenta a base de dados construída por Vilic (2014) e em seguida nas seções 4.1.2, 4.1.3 e 4.1.4 tratam do desenvolvimento realizado no escopo desta tese, detalhando o carregamento e preparação de dados, os procedimentos e os resultados respectivamente.

4.1.1 Descrição da base pública AVI SSVEP

A base possui dados EEG mensurados de participantes saudáveis, sendo expostos a alvos com frequências cintilantes para a obtenção da resposta SSVEP. Estes dados foram gravados utilizando três eletrodos (Oz, Fpz e Pz) posicionados de acordo com o sistema internacional 10-20, no qual os dados obtidos do eletrodo Oz são os únicos gravados na base de dados. O eletrodo Fpz foi utilizado como referência e o eletrodo Pz para ligação terra. Foi utilizado para a geração dos estímulos um monitor LCD BenQ XL2420T com taxa de atualização de 120 Hz. O equipamento EEG utilizado foi o g.USBamp que possui uma taxa de amostragem de 512 Hz, com eletrodos banhados a ouro, umidificados com gel eletrolítico. Os participantes tiveram que se concentrar durante o experimento em alvos de 2,89 cm² na tela do monitor, sentados a uma distância de 60 cm do alvo. A configuração geral desta experimentação é apresentada na Figura 7.

Foram realizados dois tipos de experimentos para compor esta base de dados. O primeiro foi realizado com um único alvo cintilante (*Single Target* - ST). Quatro participantes foram utilizados e cada um foi submetido a uma única sessão, focando em um único alvo durante trinta segundos, por quatro vezes (1 sessão com 4 ensaios durante 30 segundos cada). As

¹ <http://scikit-learn.org>

² <https://www.scipy.org/>

³ <http://www.setzner.com/avi-ssvep-dataset/>

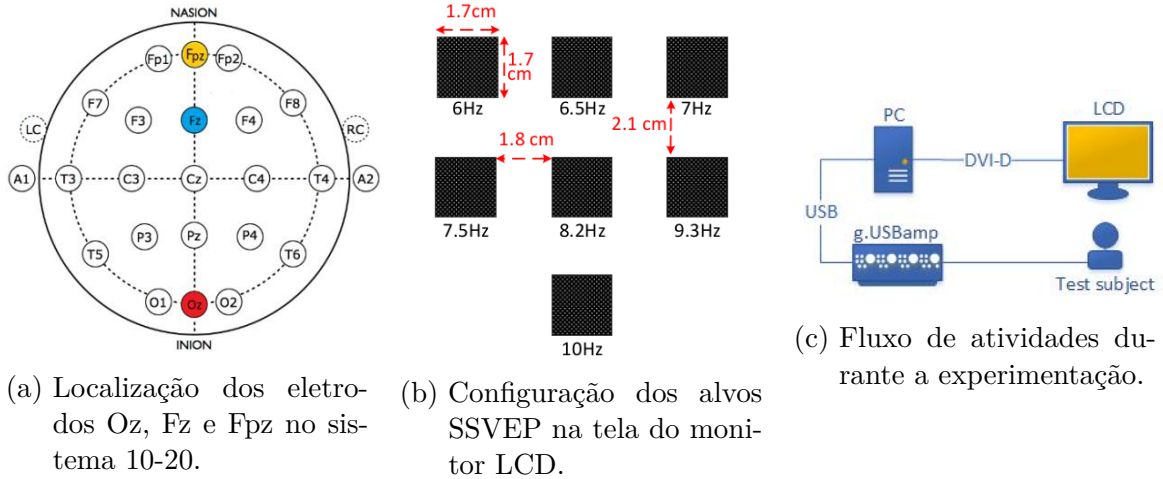


Figura 7 – Configuração geral do experimento. Figuras de (VILIC, 2014).

frequências geradas para a sequência de luzes do semáforo foram aleatórias, porém, a sequência se repetiu igualmente para qualquer seção realizada. O motivo desta experimentação com um único alvo cintilante foi para verificar a existência do sinal SSVEP. Já o segundo experimento que compõe esta base de dados, foi realizado com múltiplos alvos cintilantes (*Multi-Target* - MT), somando sete alvos em diferentes frequências. Cinco participantes foram utilizados e cada um foi submetido a duas sessões, focando em múltiplos alvos durante dezesseis segundos, por dez vezes (2 sessões com 10 ensaios durante 16 segundos cada). Em cada ensaio o participante deveria focar em um dos alvos cintilantes informado, sendo que a sequência informada também foi aleatória porém a mesma para os cinco participantes.

4.1.2 Carregamento e preparação dos dados

Foram desenvolvidos códigos para análise dos dados ST e MT. Para a análise dos resultados desta tese, será necessário ter algum parâmetro de trabalhos realizados com um único alvo, pois levando em consideração semáforos de trânsito, apenas uma luz estará acesa por vez. Por esse motivo, foi realizado a análise dos dados ST da base AVI SSVEP. Já os dados MT também foram analisados, com a premissa de avaliar um conjunto com uma maior variação de amostras, permitindo aplicar uma maior combinação de estratégias.

Originalmente os dados foram disponibilizados por (VILIC, 2014) em arquivos **MAT** (contendo os dados temporais do eletrodo **Oz** e meta-dados informativos sobre o experimento) e **CSV** (simples, somente com os dados temporais do eletrodo **Oz**). Utilizamos convenientemente os arquivos **MAT** que possui todas as informações necessárias para a experimentação e pode ser carregada inteiramente pela ferramenta **MNE**.

Nos dados ST, cada participante realizou apenas uma sessão com quatro ensaios. Porém como existem 27 ensaios em cada sessão, os dados de treino e teste puderam ser divididos

em diferentes proporções na mesma sessão, de forma que 67% das amostras (18 amostras) foram utilizadas para o treino, enquanto 33% das amostras (9 amostras) foram utilizadas para teste. Já nos dados MT, não foi dividido os ensaios da mesma seção para treino e teste do classificador, pois existem poucas amostras disponíveis, somando dez ensaios cada sessão, porém cada participante realizou duas sessões. Desta forma, foi utilizado a segunda sessão de cada participante (que possui uma maior variabilidade de classes) com dez amostras para o treino do classificador e a primeira sessão com o mesmo participante para os testes.

4.1.3 Procedimentos experimentais

Independente da divisão dos dados para cada experimento, os algoritmos para pré-processamento, extração e seleção de características e classificação foram os mesmos. O fluxo de execução e os algoritmos aplicados em cada etapa experimental é apresentado na Figura 8. Os algoritmos aplicados nos experimentos ST e MT respectivamente são apresentados nos Apêndices A.1 e A.2.

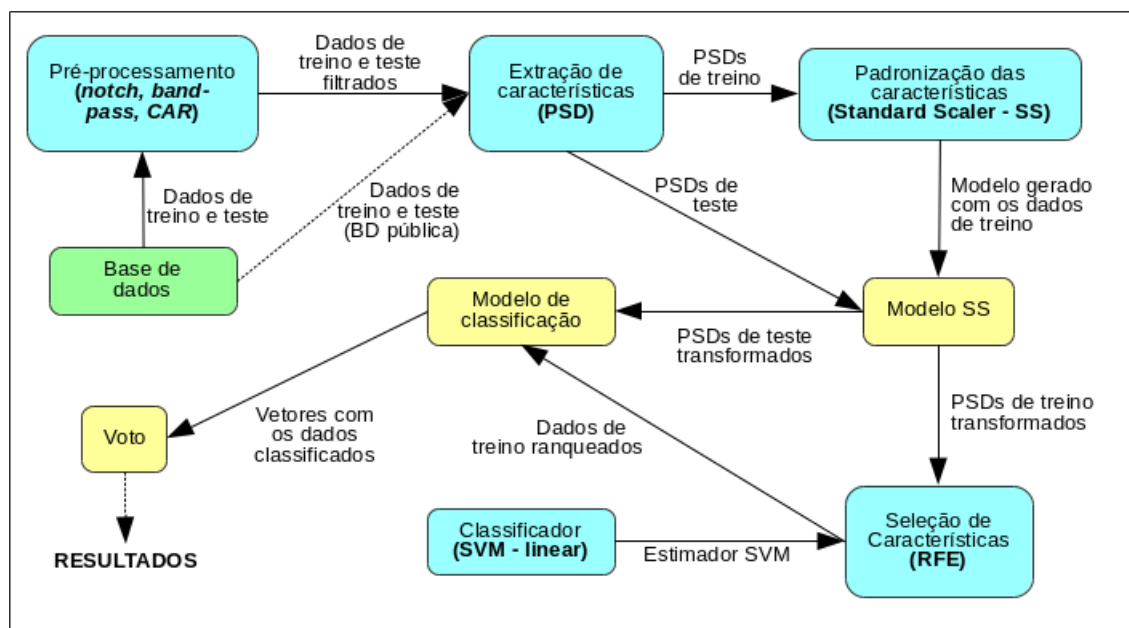


Figura 8 – Fluxo geral de execução dos experimentos apresentando os algoritmos utilizados em cada etapa.

Geralmente, o algoritmo de classificação utiliza diferentes combinações de características extraídas para o treinamento dos dados. Neste experimento, a única característica extraída é a Densidade Espectral de Potência (do inglês *Power Spectral Density* - PSD) do sinal SSVEP, que possibilita treinar o modelo de classificação independente da classe analisada. Isto ocorre pois seja qual for a frequência estimulada, espera-se ter um valor de PSD mais alto que o restante das frequências não evocadas. Sendo assim, os modelos de treinamento de qualquer frequência podem ser aplicados para classificar qualquer amostra de teste.

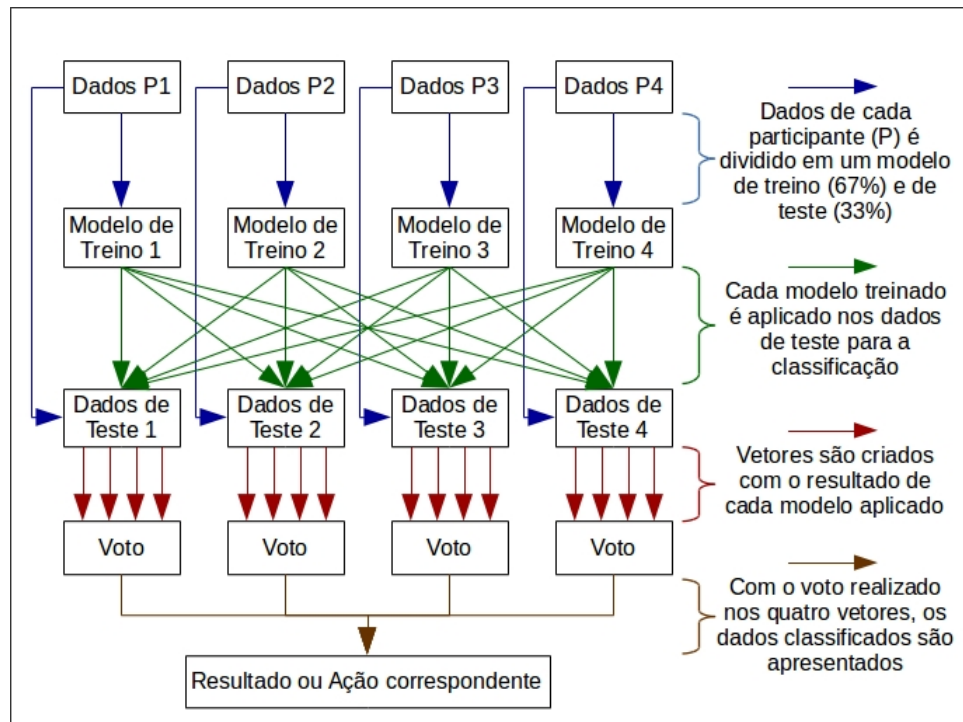


Figura 9 – Visão geral do modelo de classificação dos dados ST aplicado na base de dados AVI.

Um esquema geral da classificação dos dados ST é apresentado na Figura 9. Neste esquema, o “Modelo de Treino 1” é o modelo de classificação SVM utilizando dados obtidos do “Participante 2, 3 e 4” (setas em azul), separados para o treinamento do classificador. A mesma estratégia é utilizada para os outros Modelos de Treino. Os quatro modelos são utilizados para classificar os quatro casos de testes transformados (setas em verde) com os dados obtidos de cada participante (seta azul que vai do Participante N ao Dados de Teste N). Por fim, cada modelo aplicado na classificação irá gerar quatro vetores (setas em vermelho), no qual serão analisados pelo “Voto” que realiza a soma dos quatro vetores recebidos e obtém o máximo argumento do vetor resultante para indicar o resultado da classificação.

4.1.4 Análise dos resultados

Para a realização dos experimentos dos dados ST, foram considerados as seguintes configurações:

- Cada participante realizou a mesma sequência experimental três vezes, assim, foi possível utilizar os dados de treino e de teste em três diferentes porções divididas da seguinte forma:

1^a A primeira e a segunda sessão do experimento para treino e a terceira para teste;

- 2^a** A primeira e a terceira sessão do experimento para treino e a segunda para teste; e
 - 3^a** A segunda e a terceira sessão do experimento para treino e a primeira para teste.
- A melhor faixa de frequências para a extração de características, foi dado pelo desvio padrão igual a 0,3 (melhor valor encontrado baseado em uma execução exaustiva), ou seja, se a extração de característica foi realizada em torno de uma frequência de 6 Hz, a faixa de frequência considerada foi de 5,7 à 6,3 Hz.

O gráfico de barras com os resultados do experimento com os dados ST é apresentado na Figura 10. O melhor resultado obtido foi com o participante 4, no qual a acurácia para as três sessões realizadas foi de 100%. Já o pior resultado foi com o participante 3 utilizando a primeira sessão como teste, no qual foi obtido uma acurácia de 14%. A média geral da acurácia de todos os participantes foi de 70,75%.

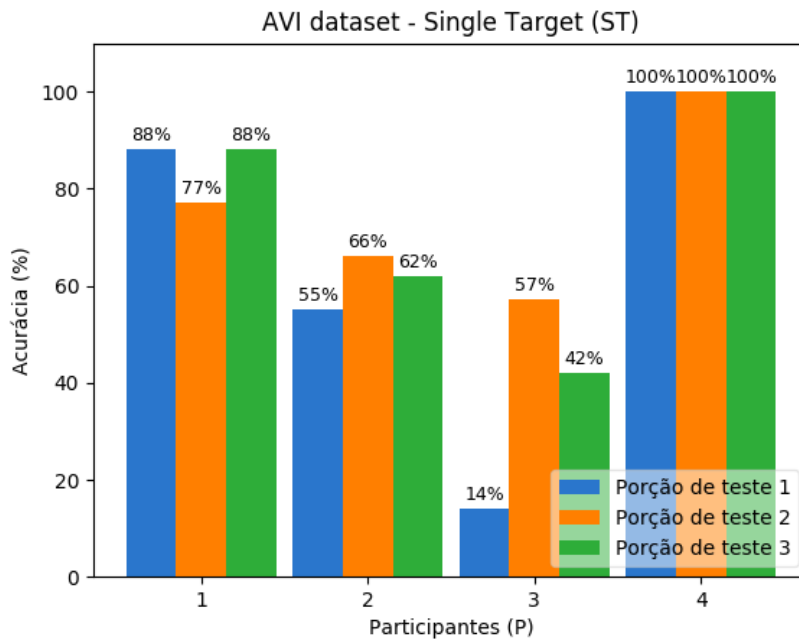


Figura 10 – Resultados do experimento com os dados *single target* da base de dados AVI.

Os gráficos de PSD foram analisados para determinar os resultados ruins apresentados pelo participante 3. A Figura 11 apresenta dois PSDs da primeira sessão. Na Figura 11a o alvo evocou um sinal de 6,0 Hz, porém o PSD é maior em torno de 12,0 Hz, já na Figura 11b o alvo evocou um sinal de 7,5 Hz, porém o PSD é um pouco maior que o evocado em torno de 10,0 Hz. Tais resultados implicam tanto no mal treinamento do classificador quanto na utilização destes dados para teste, o que resulta na baixa acurácia apresentada.

Estes gráficos foram gerados por meio do Programa 4.1. Nele utilizamos a função que gera um gráfico de PSD de todos os eletrodos utilizando o método Multi Taper, para

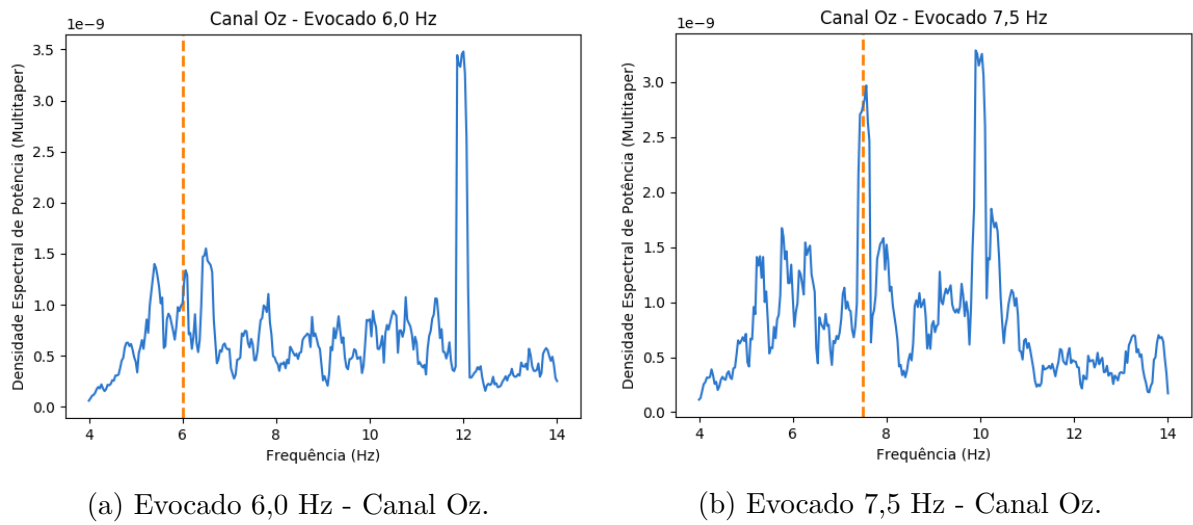


Figura 11 – Gráficos da Densidade Espectral de Potência pelo método Multi Taper, da primeira sessão realizada pelo Participante 3 no experimento com os dados *single target* da base de dados AVI.

cada exemplo da sessão com a faixa de frequência entre 4 e 14 Hz, na janela de tempo de 0 a 8 segundos. A função `view_psd_multitaper` foi desenvolvida como parte da biblioteca `bci_utils`, que é apresentada no Programa A.8 do Apêndice A.

Programa 4.1 Trecho de código que apresenta a geração do gráfico de PSD pelo método Multi Taper

```

1 # ...
2 from bci_utils import view_psd_multitaper
3 for trial in data_te):
4     view_psd_multitaper(trial, 4, 14, 0, 8)
5 # ...

```

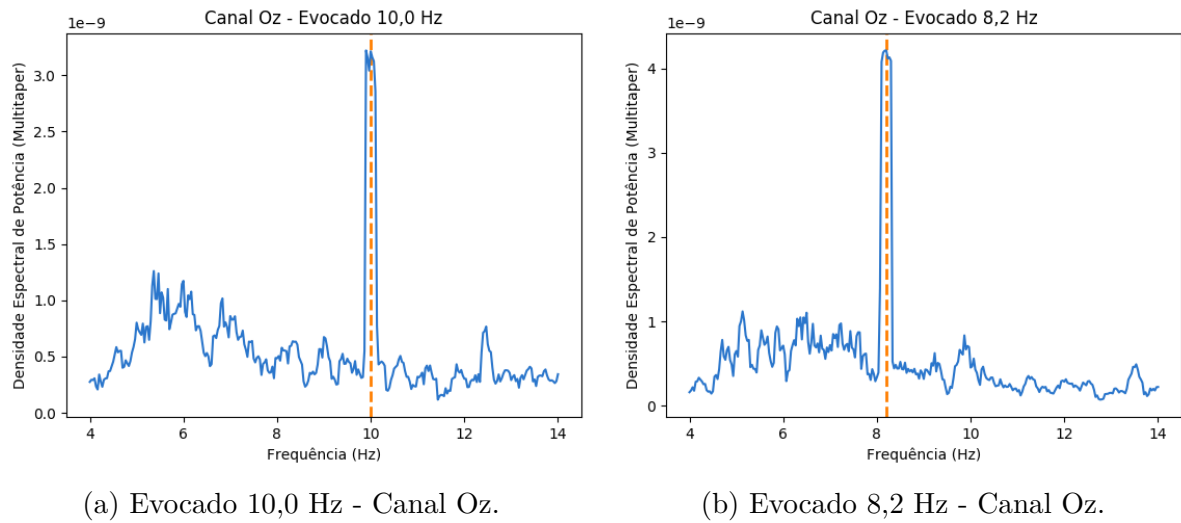


Figura 12 – Gráficos da Densidade Espectral de Potência pelo método Multi Taper da primeira sessão realizada pelo Participante 4 no experimento com os dados *single target* da base de dados AVI.

Dois PSDs da primeira sessão realizada pelo participante 4 são apresentados na Figura 12, no qual obteve a acurácia máxima de 100%. Nestas duas figuras pode ser visto que o PSD é o mais alto em torno da frequência evocada e que o restante das frequências estão com valores baixos. Estes dados influenciaram em um bom treinamento do classificador e também resultam em uma boa acurácia se utilizados para o teste.

Para a análise dos resultados com o experimento dos dados MT, foram considerados as seguintes configurações:

- Para firmar a ideia de que utilizar a segunda sessão de cada participante seria melhor pois existe uma maior variação nos dados, foram considerados para os dados de treino e de teste duas combinações diferentes:
 - 1^a A primeira sessão para treino e a segunda para teste; e
 - 2^a A segunda sessão para treino e a primeira para teste.
- A melhor faixa de frequência para a extração de características também foi com o desvio padrão igual a 0,3.

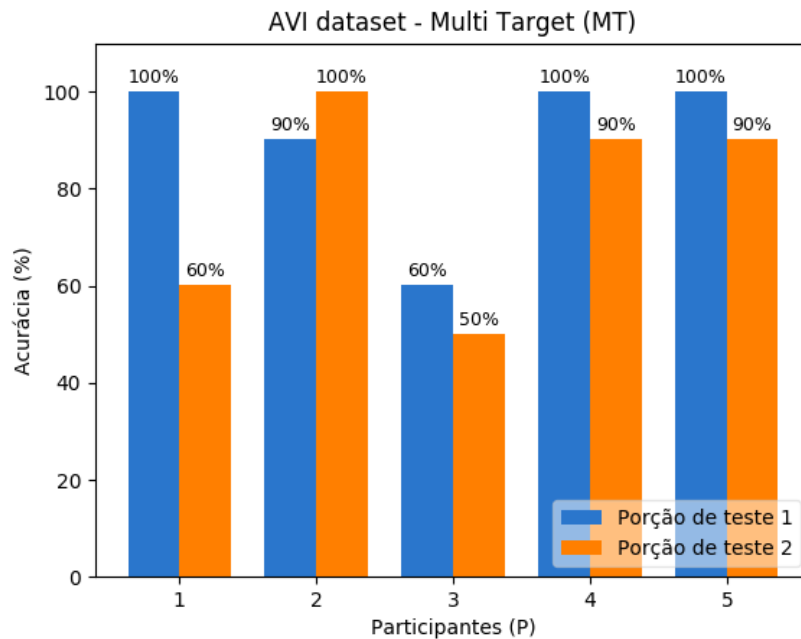


Figura 13 – Resultados do experimento com os dados *multi target* da base de dados AVI.

O gráfico de barras com os resultados do experimento com os dados MT é apresentado na Figura 13. Como era esperado, quase todos os resultados foram melhores utilizando a segunda sessão com exceção do participante 2. Os melhores resultados foram obtidos com o participante 4 e 5, no qual a acurácia foi de 100% para os dois casos utilizando os dados de treinamento da segunda sessão. Já o pior resultado foi com o participante 3 utilizando tanto os dados da primeira sessão como treinamento quanto os dados da segunda, no qual foi obtido uma acurácia de 50% e 60% respectivamente. A média geral da acurácia de todos os participantes foi de 84%.

Os gráficos de PSD foram analisados para determinar os baixos resultados apresentados pelo participante 3. A Figura 14 apresenta dois PSDs da primeira sessão realizada por este participante. Na Figura 14a o alvo evocou um sinal de 9,3 Hz, porém o PSD é maior em torno de 6,5 Hz, já na Figura 14b o alvo evocou um sinal de 6,0 Hz, porém o PSD é maior que a frequência evocada em torno de 5,0 e 7,0 Hz. Tais resultados implicam tanto no mal treinamento do classificador quanto na utilização destes dados para teste, o que resulta na baixa acurácia apresentada. O motivo de ter resultados ruins para estes participantes não pode ser afirmado, pois não temos conhecimento das condições reais do experimento.

4.1.5 Considerações finais

Os testes realizados com a base experimental de (VILIC, 2014) demonstraram que é possível utilizar os códigos desenvolvidos para avaliar um sistema SSVEP-BCI. Na próxima

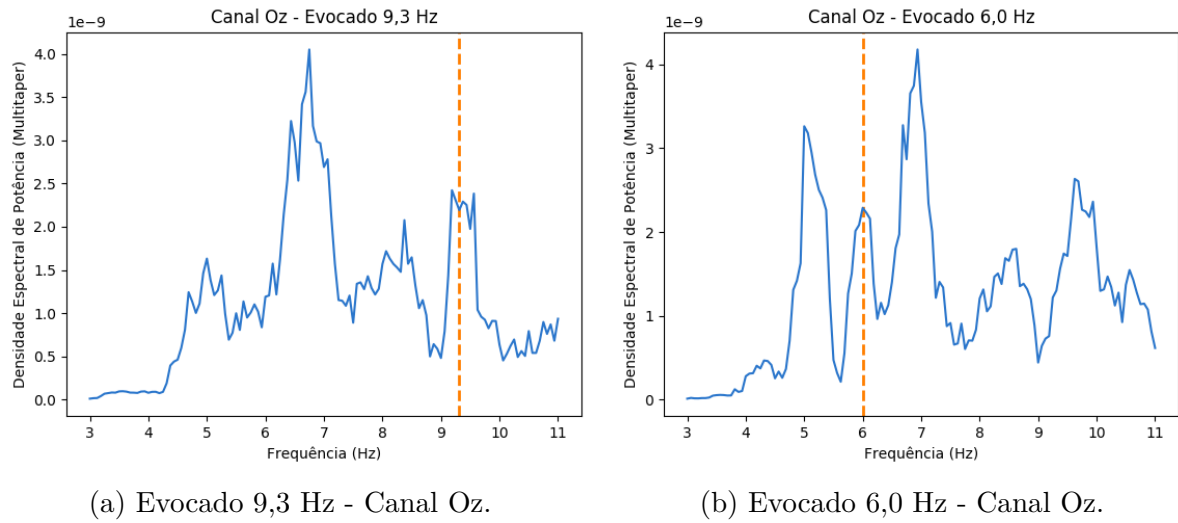


Figura 14 – Gráficos da Densidade Espectral de Potência pelo método Multi Taper da primeira sessão realizada pelo Participante 3 no experimento com os dados *multi target* da base de dados AVI.

seção será apresentado as etapas experimentais de um sistema SSVEP-BCI com um protótipo construído como parte do objetivo desta tese.

4.2 Desenvolvimento e avaliação de um sistema SSVEP-BCI com protótipos de semáforos com frequências cintilantes usando LEDs

Nesta etapa foi dado início a construção da primeira base de dados desta tese, com as informações extraídas dos experimentos com os participantes utilizando o protótipo de semáforo de trânsito. Estes experimentos colocam em prova a avaliação do protótipo utilizando semáforos com LEDs e o funcionamento do equipamento EEG utilizado. Nas próximas seções serão apresentados o protótipo, o equipamento EEG, os procedimentos experimentais, e por fim, a análise dos resultados.

4.2.1 Descrição dos equipamentos utilizados

Para a realização da simulação com o protótipo, foram construídos quatro semáforos utilizando LEDs. O semáforo construído com o restante do protótipo, construído com três LEDs difusos de 10 mm, nas cores vermelha, amarela e verde é apresentado na Figura 15a. O semáforo construído com quatro LEDs difusos de 5 mm, sendo dois nas cores vermelho, um amarelo e um verde é apresentado na Figura 15b. O semáforo construído com três LEDs alto brilho de 5 mm, nas cores vermelha, amarela e verde, é apresentado na Figura 15c. Por fim, o semáforo construído com quatro LEDs alto brilho, sendo dois nas cores

vermelho (5 mm), um amarelo (5 mm) e um verde (3 mm), é apresentado Figura 15d.

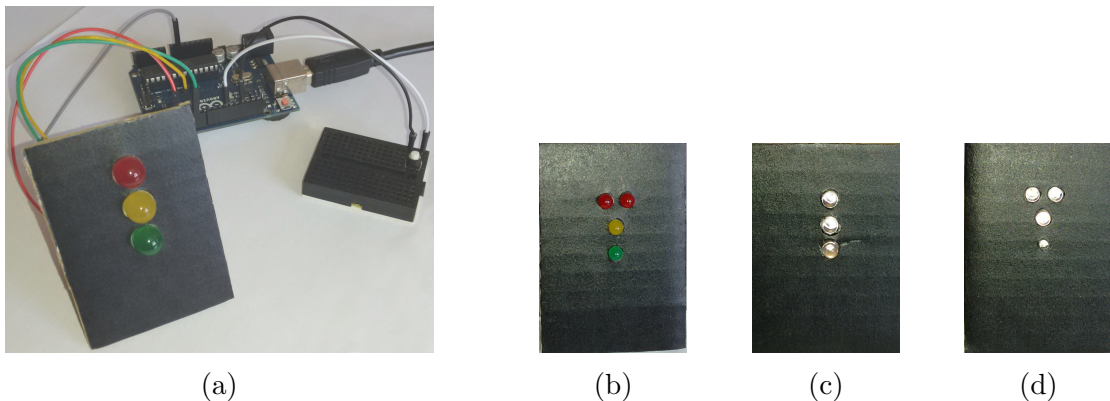


Figura 15 – Semáforos construídos com LEDs utilizados no protótipo do experimento 2.

É desenvolvida tal variabilidade de protótipos de semáforos para verificar a diferença do sinal EEG apresentada quando são utilizados LEDs difusos e de alto brilho. Os LEDs auto brilho possuem uma intensidade de luz maior apesar de causarem um incômodo visual maior que os LEDs difusos. Os semáforos que possuem dois LEDs vermelhos (tanto difuso quanto alto brilho), foram construídos para verificar se a quantidade de luz tem influência no sinal EEG quando os experimentos são realizados com semáforos não cintilantes. No experimento 2 foram utilizados apenas os semáforos (c) e (d) pois não se tem interesse de analisar a intensidade da luz, mas apenas a cintilação padrão de um sistema SSVEP-BCI.

O funcionamento dos semáforos é realizado com auxílio do Arduino UNO⁴ plataforma de prototipagem eletrônica de hardware livre, que utiliza um microcontrolador ATmega328P de 32 KB de memória *flash* e *clock* de 16 MHz. O esquema de projeto do protótipo de um semáforo com quatro LEDs é apresentado na Figura 16. Além dos LEDs ligados aos semáforos, foi também adicionado um botão momentâneo (*push button*) para controlar manualmente o início de cada sessão ou interrompê-la caso seja necessário.

O equipamento de EEG utilizado nos experimentos é o OpenBCI board⁵ de 32 bits com 8 canais para a leitura de EEG/MEG/EKG, mais 3 canais auxiliares utilizados para a leitura de um sensor giroscópico. O equipamento ainda pode ser expandido para 16 canais EEG/MEG/EKG utilizando o módulo chamado Daisy que acompanha o equipamento.

Para a realização do experimento foi utilizado um capacete desenvolvido com impressora 3D, o Ultracortex Mark 3⁶, utilizado para acoplar os eletrodos e o OpenBCI board. Os eletrodos utilizados para a experimentação são construídos com uma liga de Prata-Cloreto de Prata (Ag-AgCl), no qual dispensa a utilização de pasta ou gel eletrolítico, permitindo assim a fácil colocação do capacete em diferentes participantes durante um experimento.

⁴ <https://www.arduino.cc/>

⁵ <http://openbci.com>

⁶ https://github.com/OpenBCI/Ultracortex/tree/master/Mark_3

4.2.2 Procedimentos experimentais

Para a construção do semáforo de LEDs com frequências cintilantes, foi desenvolvido um código para o microcontrolador que permite especificar as frequências de cada LED. Este código é apresentado na Seção A.3. Em um experimento SSVEP-BCI, é desejável que múltiplos alvos tenham a cintilação em diferentes frequências, desta forma foi aplicado a seguinte equação no código do microcontrolador:

$$I = [1 / f] / 2 * 1000 - \epsilon \quad (4.1)$$

onde I é o intervalo de tempo entre as ativações do LED, f é a frequência em Hz desejada e ϵ é o atraso de *loop* do código em execução. Este atraso foi calculado utilizando um sensor de luz LDR ligado a um Arduino, no qual o sensor ficou apontado para o LED aceso em diferentes frequências e os dados lidos do sensor enviados para o computador para serem analisados por um gráfico em função do tempo. Foi verificado que este atraso varia de 1 a 2 milissegundos, então a média deste valor (1,5 milissegundos) foi atribuído para ϵ .

Neste primeiro experimento com o protótipo, as seguintes frequências para cada LED foram configuradas: **vermelho** igual a 8 Hz, **amarelo** igual a 10 Hz e **verde** igual 12 Hz. Foi escolhida frequências não múltiplas uma da outra, o que impede de ocorrer um fenômeno de sobreposição no espectrograma, fazendo com que magnitude do sinal seja alta em torno dos múltiplos da frequência evocada. Além disso, segundo Sakurada *et al.* (2015), frequências abaixo de 15 Hz são as que possuem a melhor resposta SSVEP.

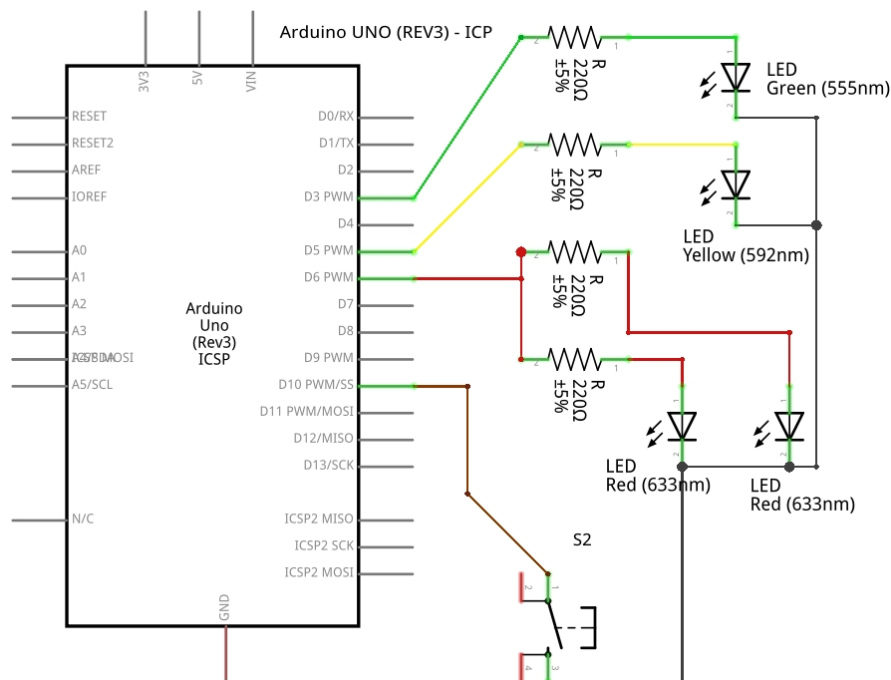


Figura 16 – Esquema elétrico de um semáforo com 4 LEDs.

Um fluxograma da experimentação detalhando o software e o hardware utilizado, bem como o modelo de comunicação realizado entre eles é apresentado na Figura 17. A obtenção do sinal EEG por meio do OpenBCI board é realizada com o software OpenBCI GUI v2⁷, que permite a visualização do sinal EEG no domínio do tempo, da frequência e ainda permite filtrar as frequências de interesse para a visualização do sinal. Este software ainda permite reenviar o sinal obtido utilizando uma comunicação por datagramas (UDP). Ao configurar o OpenBCI GUI v2 para enviar os dados pela comunicação UDP (servidor), é possível obter o sinal por outra via UDP (cliente) e tratá-lo adequadamente conforme a necessidade da aplicação. Desta forma, foi desenvolvido o código, apresentado na Seção do Apêndice A.5, no qual utiliza o UDP para receber o sinal EEG, que é gravado em um arquivo FIF (extensão do arquivo da ferramenta MNE) juntamente com os marcadores recebido pela porta serial do microcontrolador. Tais marcadores são as indicações de tempo que denotam o momento em que cada luz no protótipo de semáforo foi acesa, sendo fundamental para sincronizar com os dados de EEG.

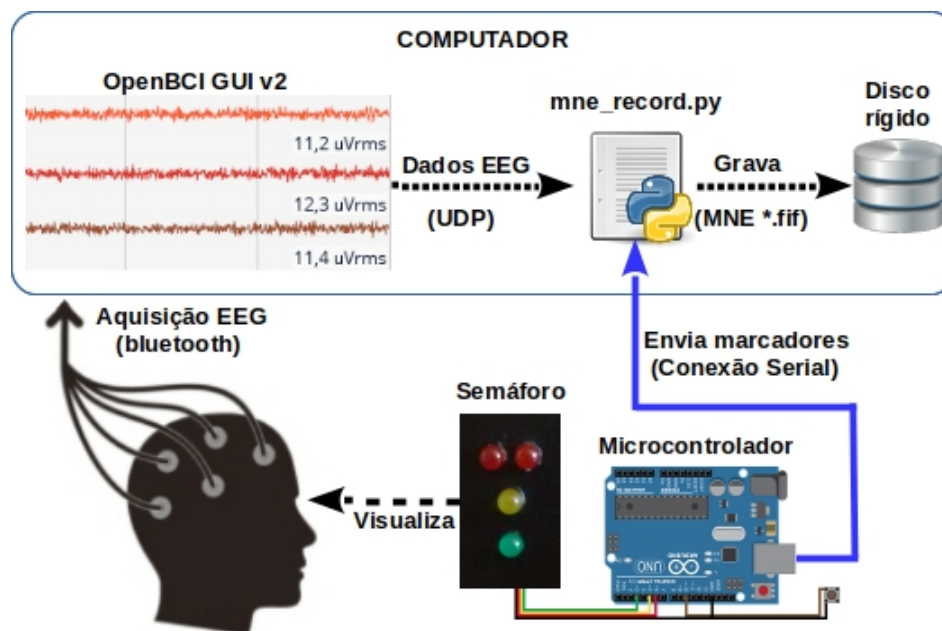
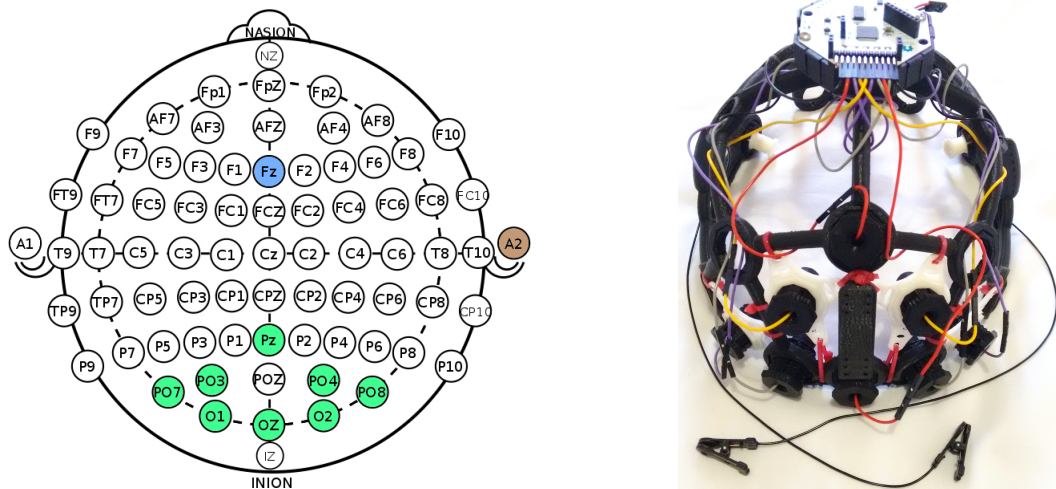


Figura 17 – Representação do fluxo do experimento 2.

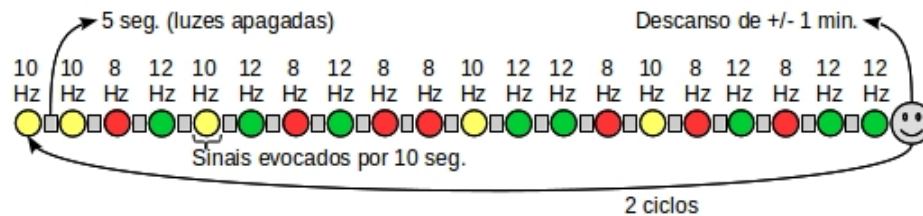
Esta etapa dos experimentos foi realizada com apenas um participante, pois o objetivo foi colocar em prova o funcionamento correto do equipamento EEG e verificar se o ambiente físico simulado com o protótipo é suficiente para evocar um bom sinal SSVEP. O seguinte protocolo para a realização das sessões foi adotado:

- O experimento foi realizado em um ambiente interno com luminosidade baixa, controlado, a ponto de poder parar o experimento a qualquer momento que o participante desejar;

⁷ https://github.com/OpenBCI/OpenBCI_GUI



(a) Configuração de eletrodos utilizada no sistema 10-20. (b) Equipamento utilizado para a leitura do sinal EEG.



(c) Etapas do ciclo experimental.

Figura 18 – Ilustrações do protocolo referente ao experimento 2.

- O participante ficou sentado aproximadamente a 1 metro de distância do alvo (protótipo de semáforo);
- O alvo utilizando LEDs de alto brilho não foi apontado diretamente nos olhos do participante, pois devido a sua alta intensidade de luz, pode causar desconforto na visão;
- O participante é exposto a duas sessões. A sequência de uma sessão é mostrada na Figura 18c. Em cada sessão foi evocado o sinal SSVEP vinte vezes com uma sequência de luzes aleatória no alvo. Durante a sessão, o LED fica ativo por 10 seg. com intervalos de 5 seg. entre uma ativação e outra. Desta forma, uma sessão tem a duração de 15 min. e 42 seg. de experimento;
- Os dados de EEG e dos marcadores foram gravados em um único arquivo FIF (referente a ferramenta MNE) em uma base de dados para posterior análise *offline*.

Para mensurar o sinal EEG no padrão SSVEP, os eletrodos foram posicionados no couro cabeludo do participante em localizações do lobo occipital, parieto-occipital e parietal, respeitando o padrão do sistema internacional 10-20. A Figura 18a apresenta as posições

dos oito eletrodos que mensuram o sinal EEG (01, 0z, 02, P03, P04, P07, P08 e Pz), mais dois eletrodos utilizados para referência e ligação terra (Fz conectado no lobo frontal central e A2 conectado no lobo da orelha direita respectivamente). Já a Figura 18b apresenta a montagem completa do equipamento OpenBCI board ligado aos eletrodos Ag-AgCl, juntamente com o capacete Ultracortex Mark 3.

4.2.3 Análise dos resultados

Para avaliação dos resultados, foram considerados apenas os dados do experimento realizado com o semáforo utilizando LEDs de alto brilho. A resposta SSVEP não ocorreu ao utilizar o protótipo de semáforo com LEDs difusos por ter uma intensidade de luminosa muito baixa. Sendo assim, será revisado o protótipo de semáforo com LEDs difusos para os experimentos futuros.

O código apresentado na Seção A.6 do Apêndice A realiza a validação e análise dos resultados do experimento 2. Algumas modificações foram realizadas neste código em relação ao utilizado no conjunto experimental 1. A primeira modificação foi em relação ao carregamento dos dados EEG, no qual estes já estavam no arquivo referente à ferramenta MNE e, além disso, foi desenvolvido e aplicado o filtro espacial *Common Average Reference* (CAR), tomando como referência os canais 01, 02, P04 e P07, pois foram os canais com maior resposta VEP nos experimentos obtidos, além de filtros temporais de passa-faixas FIR (janela de Hamming) nas frequências de corte de 5 Hz e 50 Hz e um filtro *notch* nas frequências de 60 Hz e 120 Hz.

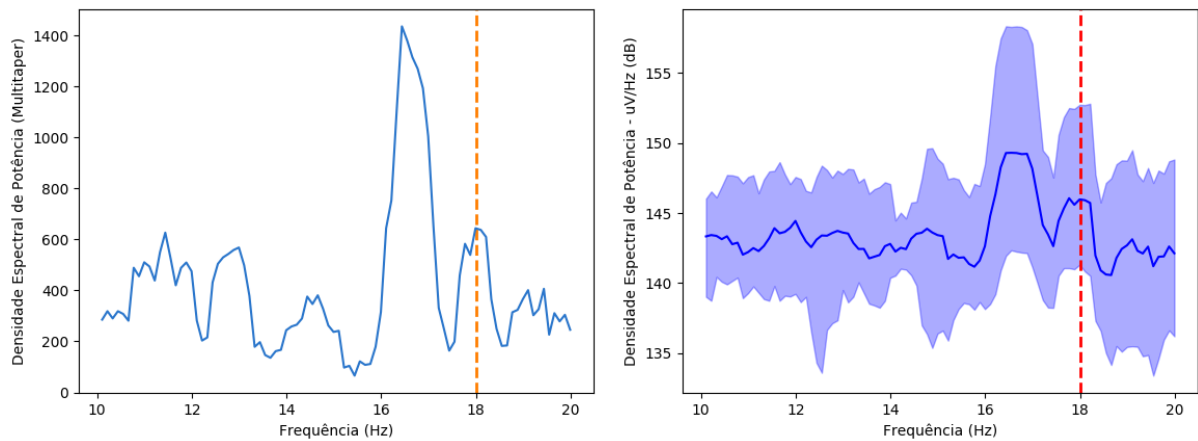


Figura 19 – Gráficos da Densidade Espectral de Potência de um sinal evocado em 18 Hz com ruído em aproximadamente 16,5 Hz.

Durante as sessões realizadas com o participante, houve problemas com relação a ruídos presentes em frequências não evocadas. Inicialmente foi proposto a seguinte sequência de

frequências de cintilação: vermelho (15 Hz), amarelo (12 Hz) e verde (18 Hz). Porém, nas sessões realizadas com os protótipos configurados nestas faixas de frequência, foram gerados ruídos em quase todas as sessões aproximadamente na faixa de 16,5 Hz. Dois gráficos de PSDs no qual o ruído está presente além do sinal estimulado em 18 Hz são apresentados na Figura 19. O gráfico de PSD de todos os eletrodos utilizando o método Multi Taper apresentado na Figura 19a e o PSD da topografia de todos os eletrodos com a média evidente em uma marcação mais intensa é apresentado na Figura 19b. Dado o ruído presente faixa de frequência de 16 Hz, foram configurados as frequências de cintilação então em 8, 10 e 12 Hz conforme já apresentado e uma sessão realizada com o participante foi analisada.

O gráfico de PSD com a topografia de todos os eletrodos com a média evidente foi gerado por meio do trecho de código apresentado no Programa 4.2. Nele utilizamos a função `view_psd_topo`, gerando um gráfico para cada exemplo da sessão com a faixa de frequência entre 4 e 14 Hz, na janela de tempo de 0 à 8 segundos. A função `view_psd_topo` foi desenvolvida como parte da biblioteca `bci_utils`, que é apresentada no Programa A.8 do Apêndice A.

Programa 4.2 Trecho de código que apresenta a geração do gráfico de PSD pelo método Multi Taper

```
1 # ...
2 from bci_utils import view_psd_multitaper
3     for trial in data_te):
4         view_psd_topo(trial, 4, 14, 0, 8)
5 # ...
```

Os dados de treino e de teste utilizados para a classificação foram divididos em porções de 70% e 30% respectivamente, realizando uma validação cruzada no qual os 70% iniciais foram utilizados (seis primeiras triagens) e o restante para teste, em seguida, da segunda até a sétima triagem para treino e assim por diante até completar 15 diferentes combinações; A melhor faixa de frequência para a extração de características foi com o desvio padrão igual a 1,0. Este valor foi encontrado utilizando uma execução exaustiva com as 30% das primeiras sessões utilizadas para treinamento do classificador SVM).

O gráfico com os resultados do experimento 2 utilizando a validação cruzada é apresentado na Figura 20, que aplica seis primeiras porções de dados para treino e o restante para teste, em seguida da 2^a à 7^a porção para treinamento e o restante para teste, fazendo isso até completar 15 diferentes combinações possíveis. O melhor resultado obtido foi com a 9^a porção de dados utilizada para o treino do classificador, no qual a acurácia foi de 100%. Já os piores resultados foram com a 8^a e 14^a porções de dados utilizadas para o treino do classificador, no qual foi obtido uma acurácia de 78% nos dois casos. A média geral da acurácia para toda a avaliação com a validação cruzada foi de 86%.

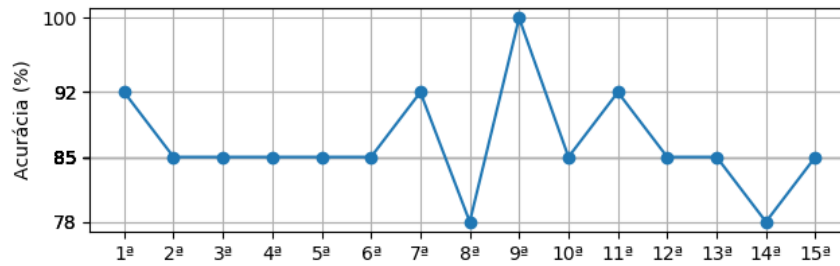


Figura 20 – Acurácia dos resultados obtidos da validação cruzada do experimento 2.

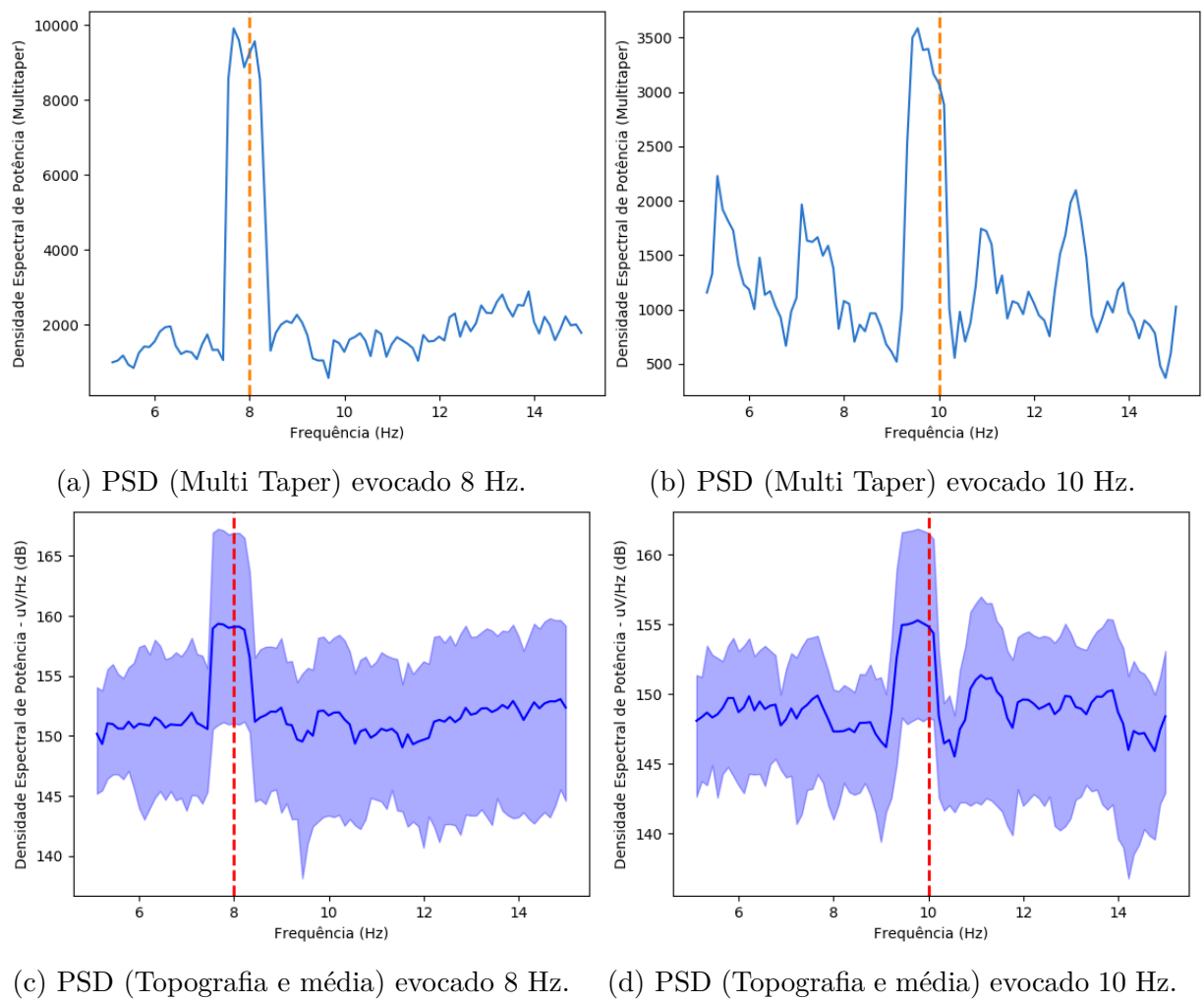


Figura 21 – Gráficos da Densidade Espectral de Potência de frequências evocados em 8 e 10 Hz.

Quatro PSDs da sessão realizada que obteve a maior acurácia (100%) são apresentados na Figura 21. Os gráficos de PSD de todos os eletrodos utilizando o método Multi Taper, com estímulos nas frequências de 8, 10 Hz, são apresentados nas Figuras 21a e 21b respectivamente e os gráficos de PSD com a topografia de todos os eletrodos com a média evidente, com estímulos nas mesmas frequências de 8, 10 Hz são apresentados nas Figuras 21c e 21d respectivamente. Pode ser observado que nas duas figuras que o máximo do PSD ocorre em torno da frequência evocada e ainda o restante das frequências possuem valores baixos. Estes dados possuem um bom treinamento do classificador e também resultam em uma boa acurácia se utilizados para o teste.

4.2.4 Considerações finais

Apesar de ruídos desconhecidos apresentados em frequências evocadas nas primeiras sessões do conjunto experimental 2, foi possível encontrar uma sequência de frequências de cintilação que fosse possível avaliar o protótipo de semáforo construído, além do equipamento de EEG utilizado para a aquisição de dados. Pretendemos investigar em trabalhos futuros causas de ruídos não esperados. No próximo capítulo serão apresentados os experimentos realizados com quatro participantes utilizando baixas e altas frequências de cintilação (visíveis e não visíveis aos nossos olhos), além de uma análise mais detalhada para viabilizar o desenvolvimento do sistema SSVEP-BCI para auxiliar na tomada de decisões.

5 Desenvolvimento e avaliação de um sistema SSVEP-BCI para auxiliar em tomada de decisões

Este capítulo apresenta o desenvolvimento, a discussão e os resultados dos experimentos realizados *offline* com baixas e altas frequências de cintilação. Foi realizada uma série de experimentos com quatro participantes com a premissa de desenvolver um sistema SSVEP-BCI para auxiliar em tomada de decisões, no qual foram avaliadas desde as frequências mais baixas até frequências altas o suficiente para que a cintilação não fosse perceptível ao olho humano e, desta forma, sendo mais próximas de situações reais de uma tomada de decisão. Além disso, frequências mais altas proporcionam um maior conforto visual aos usuários do sistema.

Os experimentos citados neste capítulo sofreram modificações nos componentes físicos e lógicos em relação aos experimentos anteriores, baseadas nos erros que foram aprendidos. Na Seção 5.1 serão apresentadas as mudanças realizadas no protótipo e na configuração do equipamento de EEG utilizado, bem como as modificações no procedimento experimental. Na Seção 5.2 serão apresentados e analisados os resultados obtidos. Por fim, na Seção 5.3 será apresentada uma discussão sobre as experimentações realizadas e as perspectivas para o desenvolvimento de um sistema BCI para tomada de decisões em situações reais. Para esta etapa da tese, adotamos o nome de **3º conjunto experimental**.

5.1 Procedimentos experimentais

Com a análise dos resultados obtidos no experimento 2 (Seção 4.2), foi observado que os protótipos de semáforos construídos com LEDs difusos não tinham efeito significativo na resposta SSVEP. Portanto, foi utilizado deste ponto em diante somente um protótipo de semáforo com LEDs alto brilho.

Com relação ao equipamento de EEG, foi realizado uma mudança em relação ao posicionamento do eletrodo de referência, que antes estava posicionado em Fz e agora está posicionado em A1 (lóbulo da orelha esquerda). Esta mudança foi realizada por dois motivos: a primeira é que foi mais fácil posicionar o eletrodo de referência no lóbulo da orelha do que no couro cabeludo com a utilização do capacete, pois como era uma posição oposta dos demais eletrodos posicionados na região occipital, parieto-occipital e parietal era mais difícil fazer com que o eletrodo tocasse no couro cabeludo do participante juntamente com os demais e a leitura de referência não era realizada adequadamente. A nova configuração

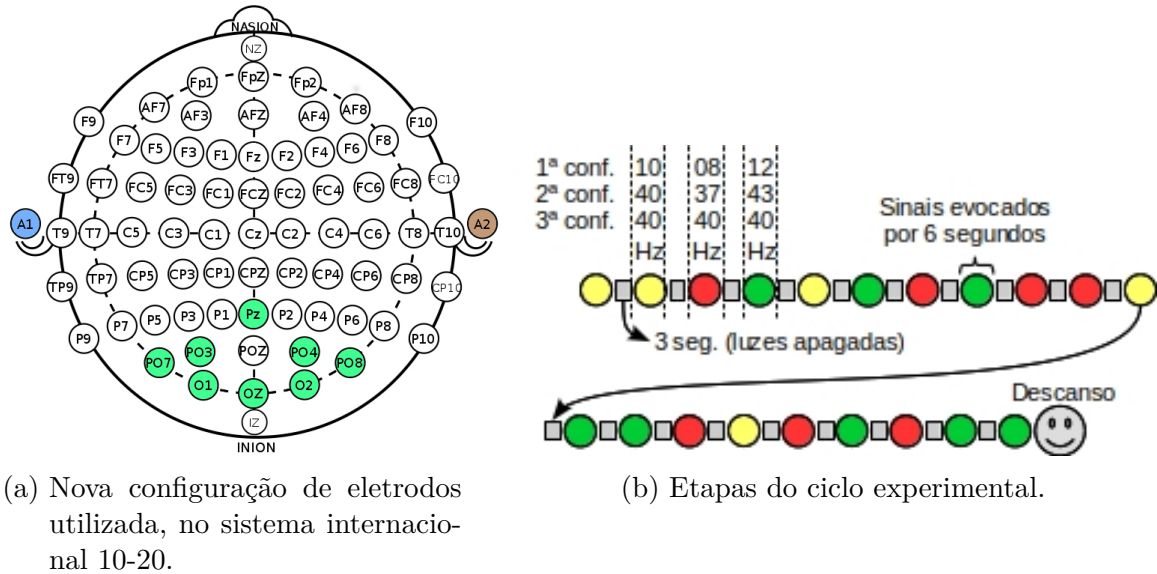


Figura 22 – Elementos modificados no protocolo referente ao 3º conjunto experimental.

das posições dos eletrodos é apresentada na Figura 22a.

Foram convidados quatro participantes saudáveis para a realização dos experimentos. Esta etapa experimental está em aprovação pelo Comitê de Ética e Cidadania, com o parecer substanciado número 2.867.988 CAAE 92010318.3.1001.5547 apresentado no Apêndice ???. O Termo de Consentimento Livre e Esclarecido (TCLE) entregue a cada participante antes da execução do experimento é apresentado no Apêndice B. Cada um participou de três diferentes experimentos com duas sessões cada um. O tempo da sessão foi reduzido em relação ao conjunto experimental 2 para não causar cansaço aos participantes. Além disso, o conjunto completo deste experimentos foi realizado em dois dias com cada participante, de modo que o 1º dia foi realizado os experimentos com baixas frequências e no 2º dia com as altas frequências de cintilação. A divisão dos experimentos em dois dias buscou evitar resultados errôneos devido ao cansaço de cada participante. De acordo com os procedimentos realizados no TCLE, ao final de cada dia foi aplicado um questionário ao participante para verificar o nível do incômodo visual. Este questionário foi importante para detectar se realmente as frequências mais altas geram menor incômodo visual que as frequências baixas.

A linha de tempo das sessões com as três diferentes configurações é apresentada na Figura 22b. O posicionamento do participante continuou sendo a 1 metro do alvo, sem que as luzes ficassem posicionadas diretamente nos olhos para reduzir o incômodo. As fotos da realização de um experimento com um dos participantes são apresentadas na Figura 23. As três diferentes configurações do conjunto experimental 3 são apresentadas como segue:

- **1ª configuração.** Foram repetidas as configurações realizadas no 2º conjunto experimental, para que fosse possível comparar o sinal SSVEP nos outros participantes e

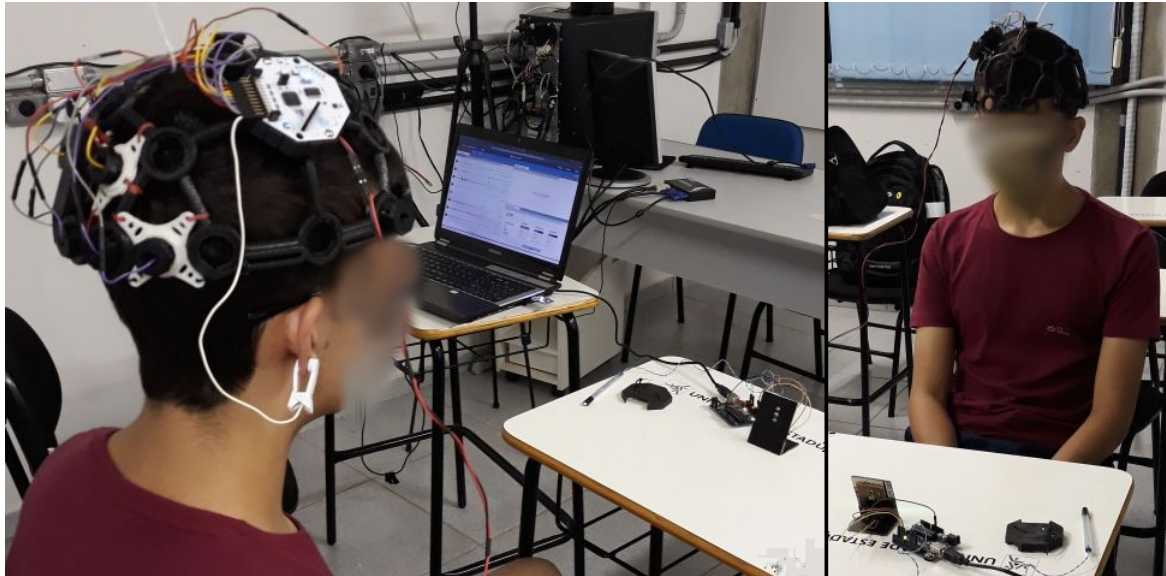


Figura 23 – Fotos da realização do 3º conjunto experimental com um dos participantes.

constar que no modelo tradicional com baixas frequências de cintilação era possível resposta adequada para estes participantes. As frequências configuradas para cada luz do protótipo foram: 8 Hz para vermelho, 10 Hz para amarelo e 12 Hz para o verde.

- **2ª configuração.** Os alvos foram configurados com diferentes frequências de cintilação, de tal forma que estas frequências não estejam visíveis ao olho humano. Os sistemas SSVEP-BCI utilizam frequências geralmente até 30 Hz no máximo, porém esta configuração tem o objetivo de considerar as altas frequências para que o modelo projetado no protótipo seja similar às situações reais de tomada de decisão. Além disso, altas frequências de cintilação geram menor desconforto visual aos participantes. As frequências configuradas para cada luz do protótipo foram: 37 Hz para vermelho, 40 Hz para amarelo e 43 Hz para o verde.
- **3ª configuração.** Nesta configuração foi utilizada a **mesma frequência de cintilação** quando os alvos estão ativos, ainda que estas frequências não sejam visíveis a olho humano. A frequência de cintilação de 40 Hz foi imposta para todos os LEDs do protótipo de semáforo. Este experimento teve como objetivo diferenciar a cor/luminância por meio da amplitude de resposta SSVEP na mesma frequência estimulada. Foi analisado o estímulo SSVEP para os três LEDs utilizando as informações da diferença de amplitude na frequência evocada. O objetivo desta configuração não foi encontrar resultados para medir os acertos (acurácia) e sim, verificar se houve diferença significativa dos estímulos SSVEP nas diferentes cores das luzes emitidas pelos LEDs. Como apoio deste experimento, o trabalho realizado por Cao *et al.* (2012), apresenta como as cores utilizadas como alvo podem influenciar no valor de fase em um sistema SSVEP em frequências que vão de 14 à 30 Hz.

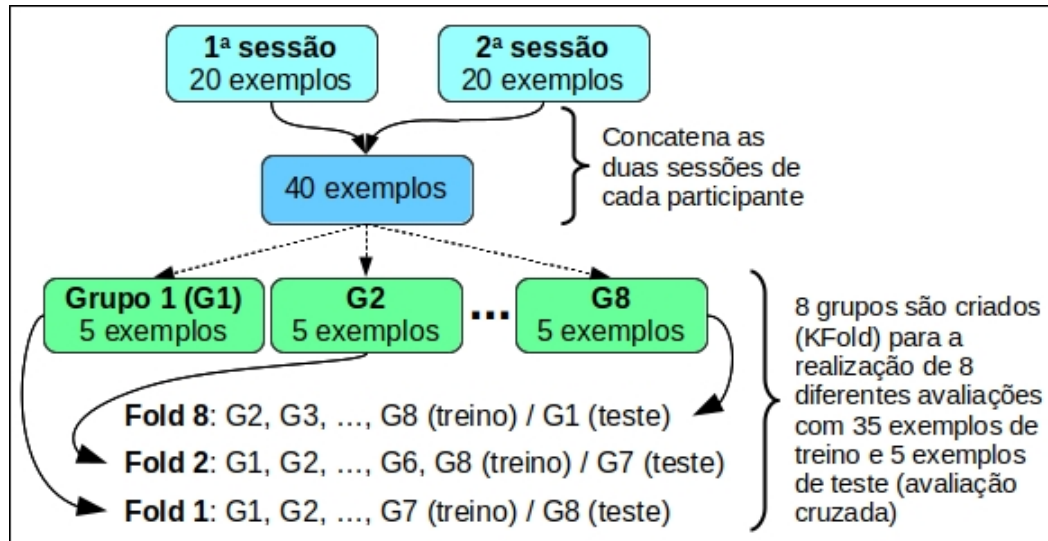


Figura 24 – Divisão dos dados resultante de duas sessões de cada participante.

Verificamos a partir destes resultados a possibilidade de utilizar como característica a resposta SSVEP de diferentes cores, para maximizar o desempenho de sistemas SSVEP-BCI baseado em altas frequências em trabalhos futuros.

Para encontrar as frequências de cintilação não visíveis foi escrito um programa para o microcontrolador, apresentado no Programa A.4 do Apêndice A, no qual diversas frequências são apresentadas a cada participante de forma crescente a partir de 30 Hz, e a cada 5 segundos a frequência aumenta de 1 em 1 Hz, até que o participante comunique que a frequência de cintilação não está mais visível. Todos os participantes comunicaram que em 40 Hz não foi possível ver o intervalo no qual o LED acende e apaga, sendo possível ver apenas uma ondulação em relação à luz refletida inerente à frequência ainda imposta no protótipo. Após o experimento, três diferentes frequências não visíveis em torno de 40 Hz foi utilizada para a 2ª configuração e para a 3ª configuração foi fixada a frequência de 40 Hz.

Na realização da análise do 3º conjunto experimental algumas alterações foram realizadas em relação à análise do 2º conjunto experimental apresentado no capítulo anterior. A mudança possibilitou um melhor aproveitamento dos dados EEG obtidos, possibilitando juntar as duas sessões realizadas por cada participante em uma só. Desta forma, como cada sessão havia 20 exemplos, juntando as duas temos 40 exemplos com 3 diferentes classes nos quais 16 representam luz verde, 14 representam a luz vermelha e 10 representam a luz amarela. A divisão resultante nestes 8 grupos possibilitou criar 8 diferentes grupos de dados (*folds*) divididos em treino e teste para avaliar. O formato que os dados de duas sessões de cada participante no mesmo experimento foram divididos é apresentado na Figura 24.

Na etapa de classificação os parâmetros da SVM foram ajustados para encontrar o melhor

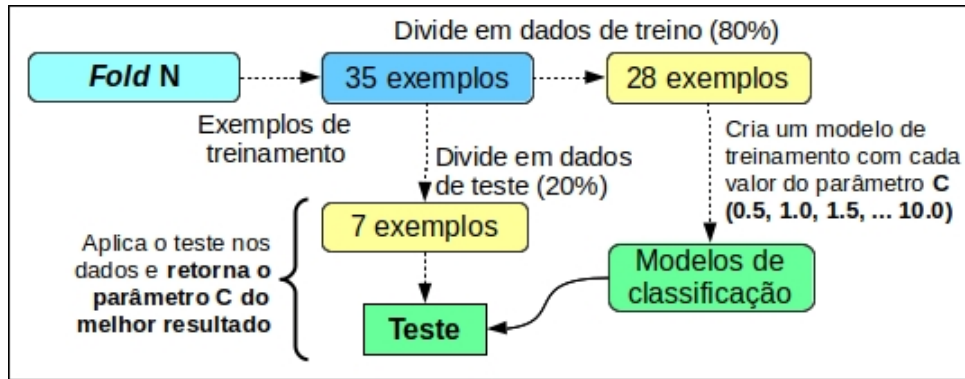


Figura 25 – Divisão dos dados para a otimização do parâmetro C do classificador SVM.

modelo de treinamento em cada configuração. As constantes C e γ (γ (gamma)) são os principais parâmetros que podem ser ajustados no modelo de treinamento do classificador SVM (explicadas previamente na Seção 2.4.4) mas de forma geral, o parâmetro C é ajustado para suavizar a margem de erros do classificador (quanto mais baixo, a margem de decisão se torna mais suave e quanto mais alto tem o objetivo de classificar todos os exemplos de treinamento corretamente) e o parâmetro γ define até onde chega a influência de um único exemplo de treinamento (valores mais baixos, resultam em uma alta influência e mais altos, uma baixa influência).

Entretanto ao utilizar a seleção de características por meio do método RFE, é necessário ter um estimador de aprendizado supervisionado no qual um atributo interno do método de classificação chamado `coef_`, que armazena valores que fornecem a importância da característica ao estimador, esteja disponível. No caso do estimador SVM, este atributo só está presente no *kernel* linear. O parâmetro γ é utilizado apenas em *kernels* não lineares e, desta forma, somente o valor do parâmetro C foi otimizado para o modelo de classificação.

Para encontrar a melhor configuração para o parâmetro C , foram realizadas interações de treinamento e teste com alguns valores atribuídos a este parâmetro. Estas interações ocorreram individualmente em cada *fold*, nos conjuntos de dados de treino, otimizando somente o conjunto de treinamento daquele *fold*. A otimização de C levou em consideração uma proporção de dados de cada *fold* em 80% de treino e 20% de teste. Foram atribuídos 8 diferentes valores para o parâmetro (0.5, 1.0, 1.5, ..., 10.0 e o treinamento foi realizado com 80% de dados do *fold* e testado com os outros 20% dos dados. O valor de C foi escolhido de acordo com a maior valor de acurácia obtida e então utilizado em seguida para o conjunto de treinamento do *fold* em questão. No fim desta análise, foi obtido então o resultado dos 8 *folds* para cada configuração de cada participante, no qual também pôde ser calculada a média geral da configuração e a variância. A realização da divisão dos dados para a otimização do parâmetro C do classificador SVM é apresentada na Figura 25.

Foi desenvolvida uma rotina de programação para encontrar uma melhor combinação dos

canais utilizados pelo filtro espacial CAR. Tentamos otimizar estes parâmetros separando intuitivamente a execução de toda a análise com cada canal separadamente, assim removendo da lista os canais que obtiveram o menor desempenho e ficando com os melhores. Por exemplo, são ao todo oito canais utilizados e experimento é executado com cada um deles conforme apresentado do Programa 5.1. Após isso, são escolhidos cinco melhores canais (maiores acurácias obtidas) no primeiro laço de repetição, por meio dos resultados apresentados (linha 12 e 13). Após isso, é executado o laço de repetição (linha X) com os cinco canais escolhidos em combinações de dois canais. Em seguida, em combinações de três canais, e assim por diante, até executar com todos os cinco. Se um dos resultados de alguma combinação encontrar 100%, o teste termina, senão ocorre até o fim. Levando em consideração que se fossemos realizar todas as combinações possíveis de todos os canais teríamos 255 diferentes. Foi desenvolvida esta abordagem para intuitivamente induzir a melhor combinação de canais em um menor tempo, realizando apenas 31 combinações no pior caso.

Programa 5.1

```

1 from itertools import combinations
2 # data_epo.ch_names : ['O1', 'Oz', 'O2', 'PO7', 'PO3', 'PO4', 'PO8', 'Pz']
3 chs = data_epo.ch_names
4 result_ch = 0 # armazena maior valor
5 comb_ch = None # armazena combinação de canal com maior valor
6 for i in range(1, 5):
7     # itera sobre todos os canais disponíveis:
8     for ch in combinations(chs, i):
9         for train_i, test_i in kf.split(data_epo):
10             # ... executa todo o processamento
11             # mostrando a média dos resultados
12             result = sum(results) / i_fold
13             if int(result) == 100:
14                 print 'Combinação', ch
15                 exit()
16             else:
17                 result_ch, comb_ch = (result, ch) if result > result_ch \
18                                     else (result_ch, comb_ch)

```

A última modificação do programa que analisa os dados do 3º conjunto experimental, tem como objetivo realizar uma análise estatística dos resultados em relação a diferentes janelamentos de tempo. Como explanado anteriormente, cada exemplo (*trial*) de uma sessão qualquer realizada pelo participante, possui 6 segundos de duração. Porém, em uma situação real de decisão potencialmente de risco, não podemos levar em consideração que uma pessoa demora até 6 segundos para tomar uma decisão. Desta forma, foi comparada a avaliação de cada configuração de um participante com janelamentos dos primeiros 2, 3, 4, 5 e 6 segundos. Tomamos como base (hipótese nula) a janela de 6 segundos e uma significância estatística utilizando o **Test T de Student** é aplicada para as outras janelas. Tradicionalmente, resolução do PSD é melhor quanto maior for a janela no domínio do

tempo. Isto geralmente ocorre pois o sinal evocado em uma determinada frequência de cintilação em que o participante está focado, se torna mais abrupta do que as outras frequências não evocadas a media que o tempo passa. Desta forma, foi então verificado se o valor-p é **maior ou igual** a 5%, que nos dá a informação de que se diminuirmos a janela de tempo em resposta ao PSD, o sistema ainda continuará consistente em tomar decisão correta.

O código de análise do 3º conjunto experimental com todas as mudanças discutidas nesta seção é apresentado no Apêndice A por meio do Programa A.7.

5.2 Análise dos resultados

Esta Seção apresenta os resultados obtidos pelo 3º conjunto experimental. As três configurações foram divididas em subseções para maior localização dos resultados.

5.2.1 1ª configuração

Os primeiros resultados apresentados é referente à 1ª configuração do 3º conjunto experimental. O gráfico de caixas da distribuição das acurácias obtidas pelo classificador para os quatro participantes é apresentado na Figura 26. Estes resultados são referentes apenas à janela máxima de 6 segundos, com a intenção de verificar o máximo desempenho que o sistema pode ter com a configuração de baixas frequências.

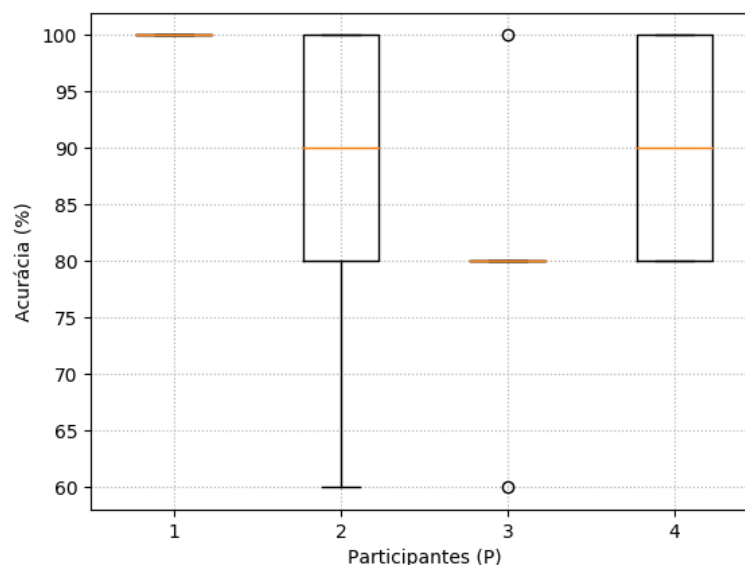


Figura 26 – Distribuição das acurácias obtidas na 1ª configuração pelos participantes na janela máxima de tempo de 6 segundos.

Tabela 1 – Combinações de canais utilizados em CAR e a média obtida por cada participantes na janela de tempo de 6 segundos da 1ª configuração

Participantes	1	2	3	4
Canais utilizados (CAR)	PO7, O2, Oz	PO7, O2	PO3	Oz, PO3, PO4
Média da acurácia (folds) (%)	100%	87%	80%	90%

A distribuição dos dados é referente aos resultados obtidos por cada *fold*. No geral, os resultados foram bons considerando que a maior parte da distribuição ficou acima de 80%. Os resultados para o Participante 1 foi o melhor, considerando que o resultados de todos os *folds* foi de 100% e o pior resultados foi para o Participante 3, no qual a média e mediana foram de 80% considerando uma variação de dados de 60 a 100%. Em média, o resultado para todos os participantes foi de 89,25%. Tais resultados foram maximizados com a melhor combinação dos canais para o parâmetro do filtro espacial CAR. Esta “otimização” dos canais foi aplicada somente para a janela de 6 segundos e então a configuração obtida foi repetida para os janelamentos de tempo menores. As combinações de canais utilizados pelo filtro espacial CAR e a média obtida por cada participante são apresentadas na Tabela 1.

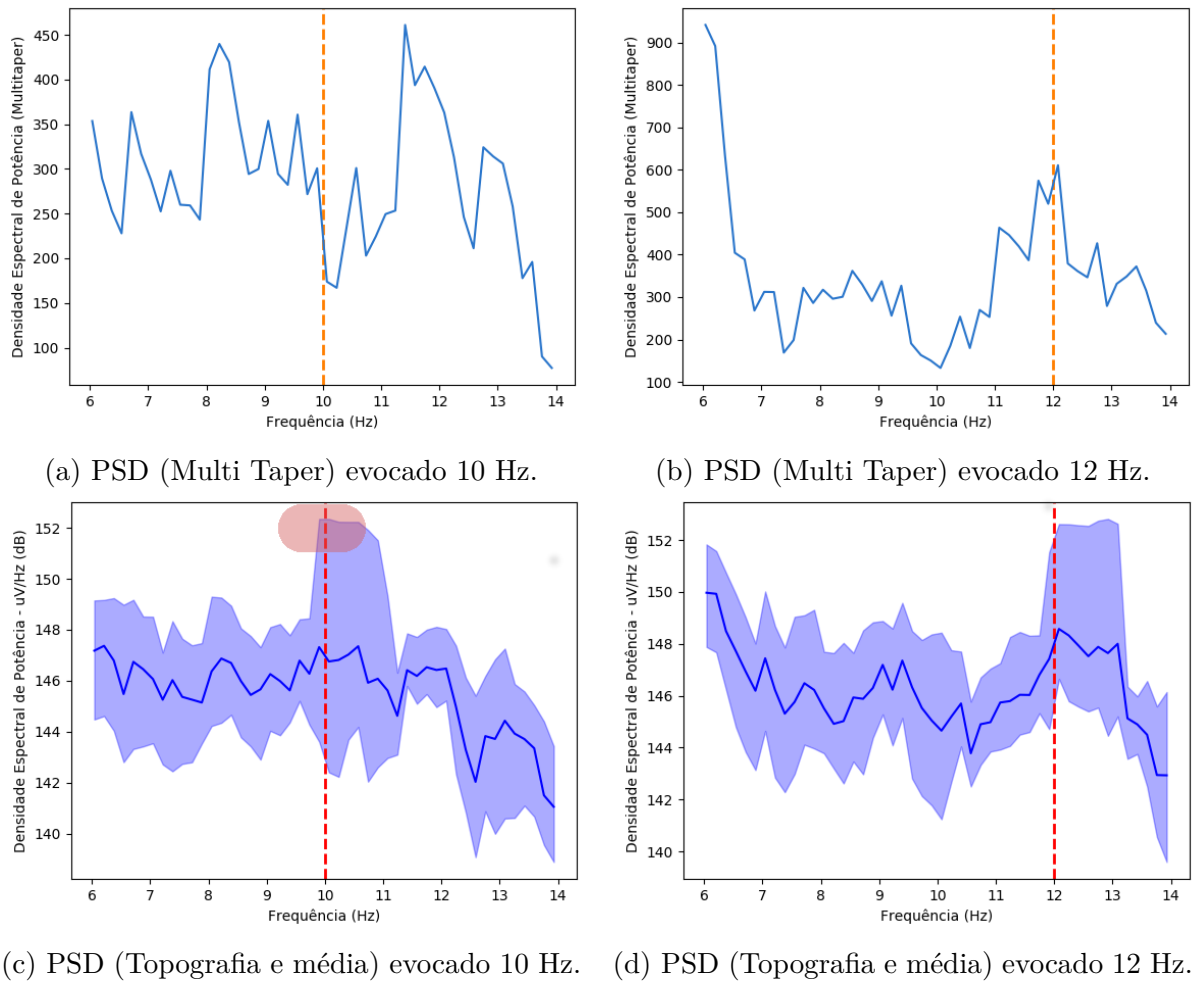


Figura 27 – Gráficos da Densidade Espectral de Potência de frequências evocados em 10 e 12 Hz.

Os PSDs da 1ª configuração nas frequências de 10 e 12 Hz, no qual obteve a maior acurácia (100%) são apresentados na Figura 27. Os gráficos de PSD com a topografia de todos os eletrodos com a média evidente são apresentados nas Figuras 27c e 27b e os gráficos de PSD de todos os eletrodos utilizando o método Multi Taper são apresentados nas Figuras 27c e 27d. Observamos que o PSD é o mais alto em torno da frequência evocada e ainda o restante das frequências possuem valores baixos. Além disso, nota-se a diferença do PSD apresentado na Figura 27a e 27c, em relação a frequência de 10 Hz evocada, pois se levarmos em consideração a média dos canais, não é visível que a frequência evocada é a de 10 Hz. Porém, quando a resposta de todos os canais, é notável a frequência de 10 Hz com o valor de PSD mais alto no limite superior do gráfico (região hachurada em vermelho).

A distribuição dos dados de acurácia referente aos resultados das janelas de tempo de 2 à 5 segundos de cada participante é apresentada na Figura 28. Podemos perceber que para a maioria dos gráficos de caixa, a média e a mediana da distribuição dos dados aumentam conforme o tamanho da janela de tempo também é aumentada. Estes resultados são

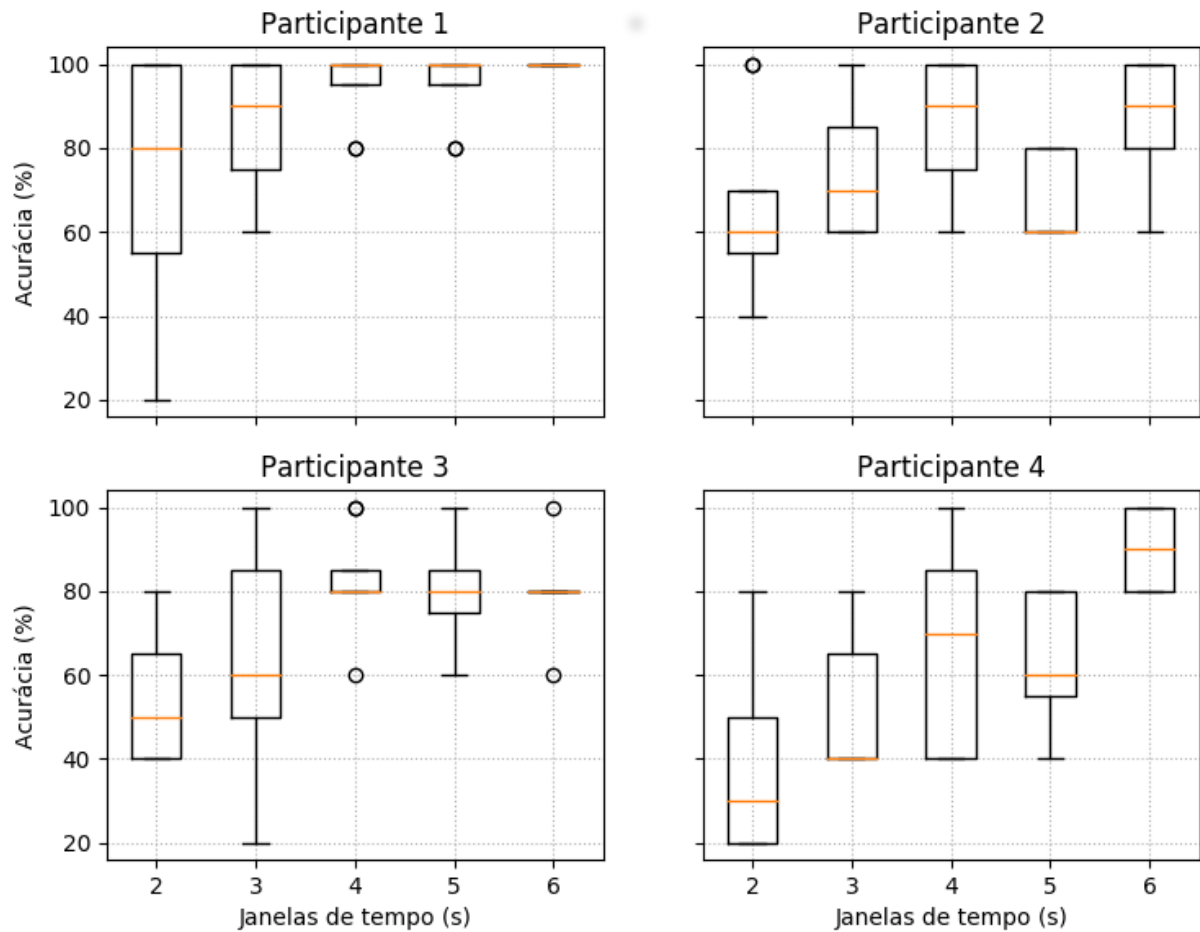


Figura 28 – Distribuição das acurácias obtidas pelos participantes em todas as janelas de tempo para baixas frequências.

esperados, já que quanto menor janela de tempo, menor é a quantidade de dados espectrais que podem ser analisados, consequentemente baixando a acurácia obtida pelo classificador.

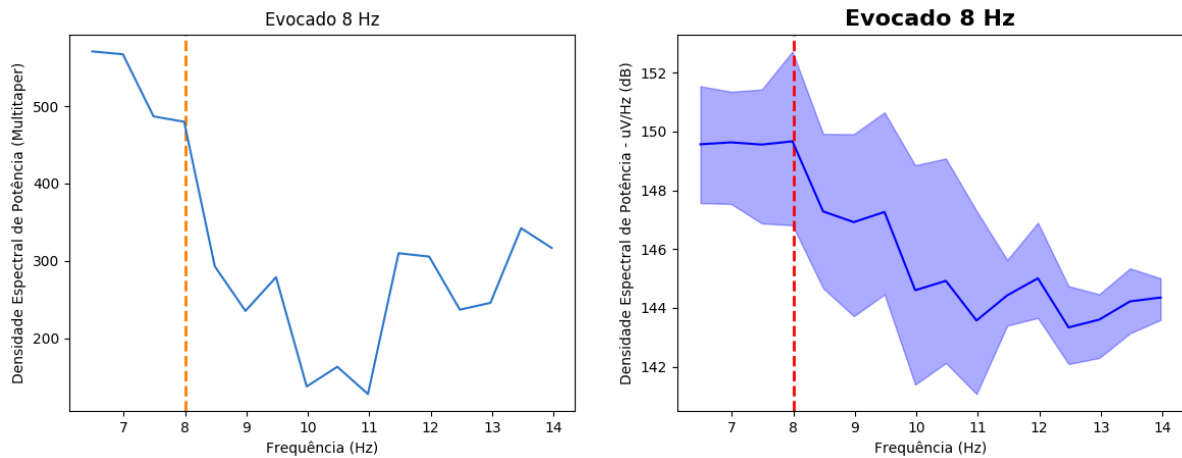
A análise do **Teste T de Student** foi realizada para os resultados de janelamento de tempo, para avaliar se o aumento do tempo para cada participante foi significativo. Lembramos que para este método estatístico, o valor de p deverá ser menor que 5% para dizer que a diferença de desempenho entre os dois sistemas é significativa. A média para cada participante (P) com o desvio padrão de cada janela de tempo e o valor de p confrontando com a maior janela de tempo (6 segundos) utilizada como base na análise, é apresentada na Tabela 2.

Podemos comprovar por meio desta análise que os resultados dos participantes 1 e 3 não obtiveram um caimento no desempenho, comparado até a janela de tempo de 4 segundos. Já os resultados de p do participante 4 demonstraram que o decréscimo no janelamento implicou em um pior desempenho para todas as comparações. Por fim, detectamos um comportamento não esperado nos resultados do participante 2, no qual o desempenho foi ruim em relação ao janelamento de 5 segundos, porém volta a ser bom em relação aos janelamentos de 4 e 3 segundos.

Tabela 2 – Significância estatística em valor-p para os janelamentos de tempo de cada participante, tomando como base a janela máxima de 6 segundos

	P1	P2	P3	P4
6 seg.	100.00% \pm 0.00%	87.50% \pm 14.88%	80.00% \pm 10.69%	90.00% \pm 10.69%
6-5 seg.	95.00% \pm 9.26% p = 14.89%	67.50% \pm 10.35% p = 0.75%	80.00% \pm 15.12% p = 100.00%	62.50% \pm 16.69% p = 0.15%
6-4 seg.	95.00% \pm 9.26% p = 14.89%	85.00% \pm 17.73% p = 76.45%	82.50% \pm 12.82% p = 67.83%	67.50% \pm 26.05% p = 4.03%
6-3 seg.	85.00% \pm 17.73% p = 3.13%	75.00% \pm 17.73% p = 14.89%	62.50% \pm 31.05% p = 15.40%	52.50% \pm 18.32% p = 0.02%
6-2 seg.	72.50% \pm 31.96% p = 2.89%	65.00% \pm 23.30% p = 3.72%	55.00% \pm 17.73% p = 0.42%	40.00% \pm 26.19% p = 0.02%

É importante destacar que os resultados obtidos com os janelamentos de dois segundos, não puderam ser bem aproveitados, pois devido à taxa de transferência do equipamento EEG ser baixa (250 Hz), este tamanho de janelamento de tempo obtém aproximadamente 500 amostras de dados no domínio da frequência (de PSD), que ainda sofrem com a aplicação do método CAR, diminuindo a quantidade de dados e impedindo que seja gerado uma quantidade de amostras suficiente para uma boa estimativa do classificador.



(a) PSD (Multi Taper) evocado 8 Hz.

(b) PSD (Topografia e média) evocado 8 Hz.

Figura 29 – Gráficos da Densidade Espectral de Potência de frequências evocados em 8 Hz em um janelamento de 2 segundos.

Dois gráficos de PSD com um janelamento de 2 segundos, de todos os eletrodos utilizando o método Multi Taper e com a topografia de todos os eletrodos com a média evidente são apresentados na Figura 29. É possível perceber apenas alguns pontos para projetar as curvas de PSD. O janelamento de 1 segundo não foi realizado, pois além da pequena quantidade de dados, a banda de transição não atende à quantidade de amostras disponíveis para a aplicação dos filtros temporais, impedindo a estimativa do PSD.

5.2.2 2ª configuração

Assim como a 1ª configuração, a 2ª configuração foi realizada com diferentes frequências de cintilação, porém mais altas do que as praticadas tradicionalmente em sistemas SSVEP-BCI.

A primeira avaliação realizada na 2ª configuração do 3º conjunto experimental refere-se também ao maior janelamento de seis segundos de todos os participantes. O gráfico de caixas da distribuição das acurácias obtidas pelo classificador para os quatro participantes é apresentado na Figura 30.

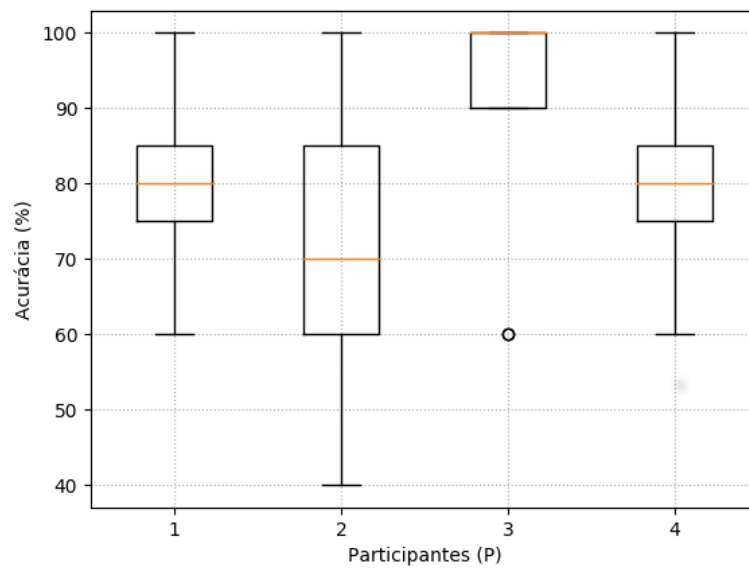


Figura 30 – Distribuição das acurácias obtidas na 2ª configuração pelos participantes na janela máxima de tempo de 6 segundos.

Como citado anteriormente em (SAKURADA *et al.*, 2015), a avaliação do sistema SSVEP-BCI em frequências acima de 30 Hz, tende a ter um menor desempenho em relação às frequências abaixo dela. Para o janelamento máximo, os resultados referente à distribuição das acurácias, podem ser considerados bons, pois na média de todos os participantes, apresentaram uma acurácia de 80,5%. Em média, os resultados individuais dos quatro participantes, obtiveram acurácias de 80%, 72%, 90% e 80% respectivamente. A distribuição dos resultados de acurácia ficaram em sua maioria acima de 80%, porém com alguns valores discrepantes muito baixos. Os resultados para o participante 1 apresentaram resultados máximos de 100% e mínimo de 60%. Quanto ao participante 3 pode ser considerado os melhores resultados, mesmo possuindo um dado discrepante de 60%. O participante 2 foi o que sofreu maior variação de distribuição, indo de 40% à 100%, o qual pode ser considerado o pior resultado. Tais resultados também foram maximizados com a melhor combinação dos canais para a parametrização do filtro espacial CAR. As combinações encontradas para os canais utilizados pelo filtro espacial CAR e a média obtida por cada

Tabela 3 – Combinações de canais utilizados em CAR e a média obtida por cada participantes na janela de tempo de 6 segundos da 2ª configuração

Participantes	1	2	3	4
Canais utilizados (CAR)	Pz, PO4	PO7, O2	PO4	Oz
Média da acurácia (folds) (%)	80.0 %	72.0 %	90.0 %	80.0 %

participante são apresentadas na Tabela 3.

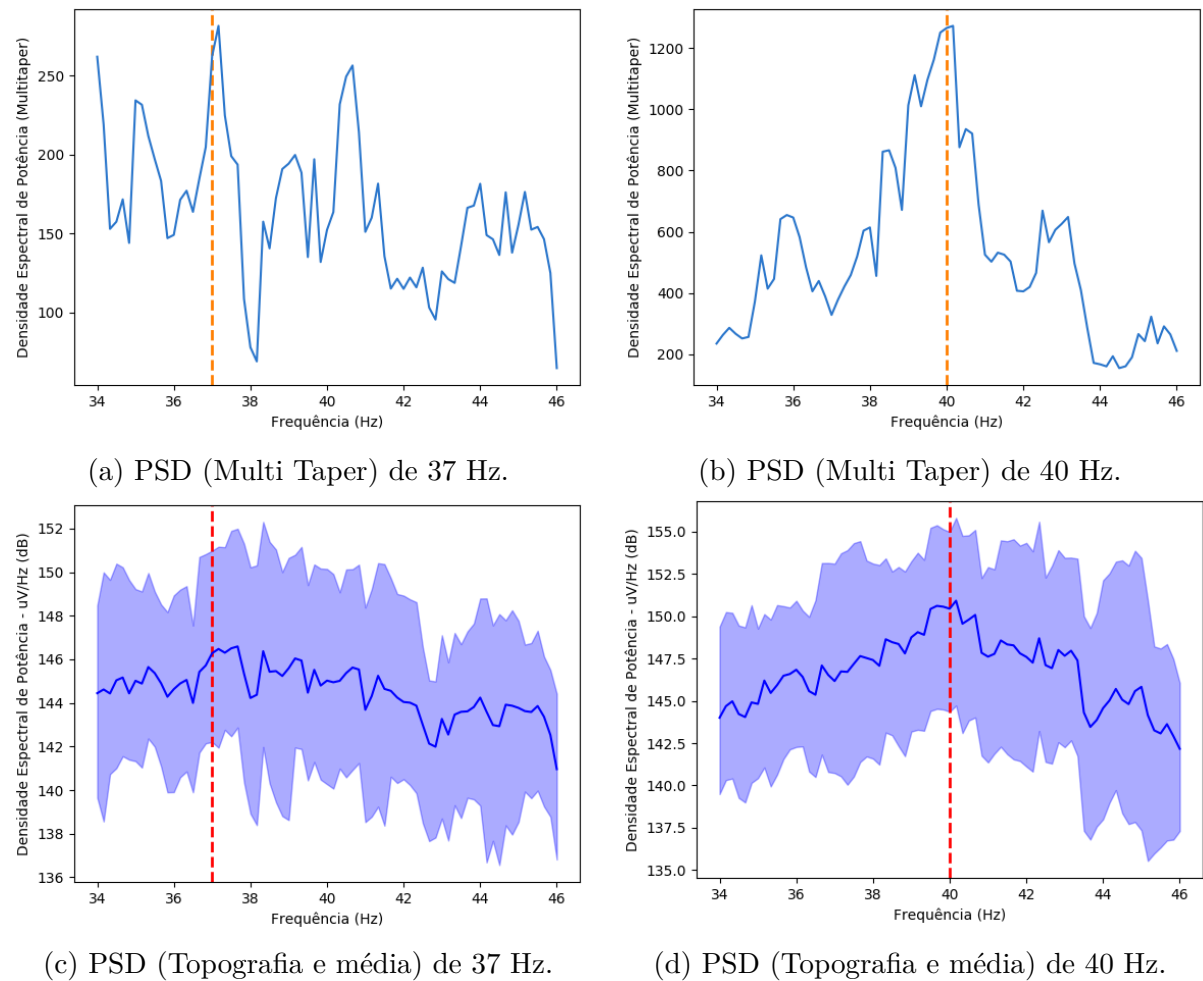


Figura 31 – Gráficos da Densidade Espectral de Potência de frequências evocados em 37 e 40 Hz.

Foram analisados alguns gráficos de PSD para o entendimento de resultados que implicam o desempenho da 2ª configuração. Os gráficos de PSD nas frequências de 37 e 40 Hz, no qual obteve uma acurácia média de 90% são apresentados na Figura 31. Os gráficos de PSD de todos os eletrodos utilizando o método Multi Taper são apresentados nas Figuras 31a e 31b e os gráficos de PSD com a topografia de todos os eletrodos com a média evidente são apresentados nas Figuras 31c e 31d. Pode ser observado que nas duas figuras referentes a topografia, o PSD é o mais alto para alguns eletrodos em torno da frequência

evocada e ainda o restante das frequências possuem valores baixos. porém, para esta 2ª configuração também nota-se que esta diferença entre os picos em torno das frequências evocadas e não evocadas não é tão evidente quanto nos gráficos vistos na 1ª configuração.

A piora no desempenho da 2ª configuração fica mais clara quando são analisados alguns gráficos de PSD que não evidenciam a frequência evocada. Dois gráficos de PSD que demonstram mal comportamento da frequência evocada são apresentados na Figura 32, na qual o PSD de todos os eletrodos utilizando o método Multi Taper na frequência evocada de 43 Hz é apresentado na Figura 32a e o PSD com a topografia de todos os eletrodos com a média evidente da mesma frequência evocada é apresentado na Figura 32b.

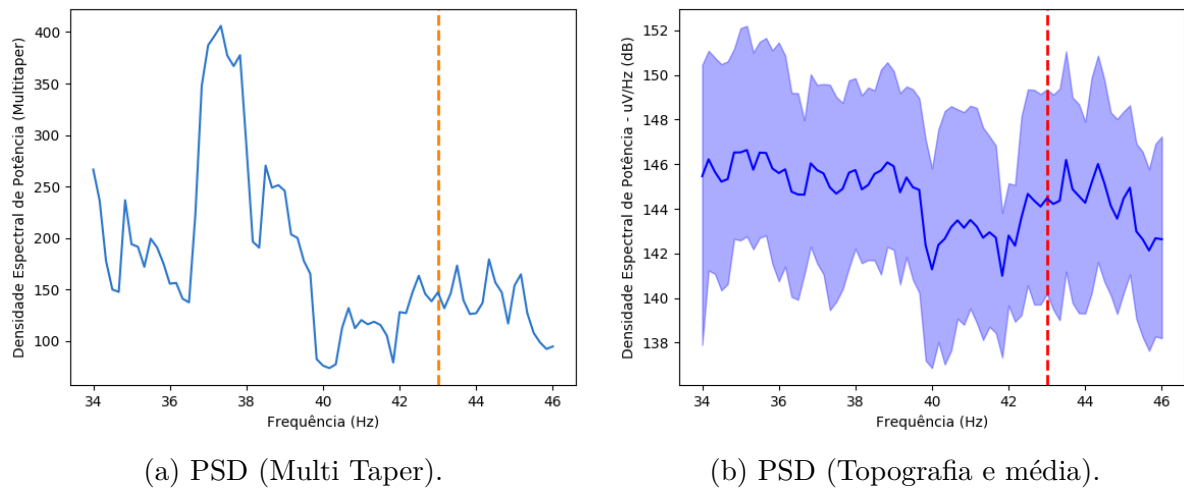


Figura 32 – Gráficos da Densidade Espectral de Potência de frequência evocada em 43 Hz.

Fica evidenciado principalmente pela Figura 32a que os maiores picos de PSD em torno de 37 Hz não condizem com a frequência evocada na frequência na frequência de 43 Hz. A maior parte dos gráficos resultantes de PSD possuem um comportamento não esperado, porém, na avaliação quando é aplicada uma otimização para a combinação de canais utilizados no filtro espacial CAR, os eletrodos que respondem de forma correta à frequência evocada acabam sendo os escolhidos, buscando assim melhores resultados.

A distribuição dos dados de acurácia referente aos resultados das janelas de tempo de 2 à 5 segundos de cada participante é apresentado na Figura 33. Nota-se nos gráficos que a distribuição dos resultados de acurácia das janelas de tempo abaixo de cinco segundos, que o desempenho foi bem inferior em comparação com a janela de seis segundos. Para os participantes 1 e 4, a média e a mediana da distribuição dos dados aumentam sua proporção conforme o tamanho da janela de tempo também é aumentada. Porém para os participantes 2 e 3 a distribuição dos resultados da acurácia estão desproporcionais.

Até então, a busca intuitiva para otimizar a parametrização de canais do filtro espacial

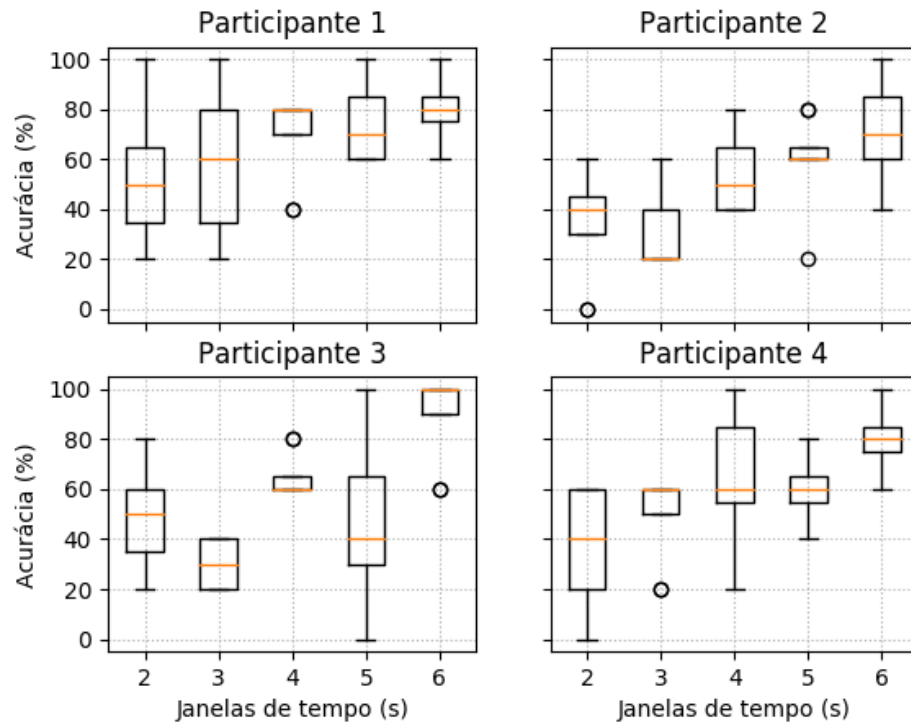


Figura 33 – Distribuição das acurácias obtidas pelos participantes em todas as janelas de tempo para altas frequências.

CAR, foi aplicada somente na janela de 6 segundos. Por convenção, a mesma combinação de canais foi utilizada para as janelas de tempo de 2 à 5 segundos, pois o sinal EEG analisado era da mesma sessão e acreditava-se que outras combinações de canais não iriam surtir uma diferença significativa. Porém, diante da desproporcionalidade dos resultados apresentados na Figura 33, foi realizado um experimento para buscar a combinação de outros canais a serem utilizados no filtro CAR para os outros janelamentos de tempo. Os resultados da distribuição da acurácia para este experimento são apresentados na Figura 34.

Nota-se pela Figura 34 que quando são otimizados os canais para todas as janelas de tempo, a distribuição das acurácias como um todo melhoram o desempenho, evidenciando assim que o desempenho é afetado positivamente quando são otimizados outras combinações de canais para diferentes janelas de tempo de uma mesma sessão.

A análise do **Teste T de Student** foi realizada para esta configuração para os resultados de janelamento de tempo. A média para cada participante (P) com o desvio padrão de cada janela de tempo e o valor de p confrontando com a maior janela de tempo (6 segundos) utilizada como base na análise, é apresentada na Tabela 4. A tabela também apresenta a coluna “P3 (CAR)” para relacionar os resultados apresentados com o participante 3, otimizando os canais para o filtro CAR em todas as janelas de tempo.

Nesta tabela, nota-se que o participante 1 não sofreu uma queda significativa no desempenho até a comparação com a janela de tempo de 3 segundos. Já os resultados apresentados

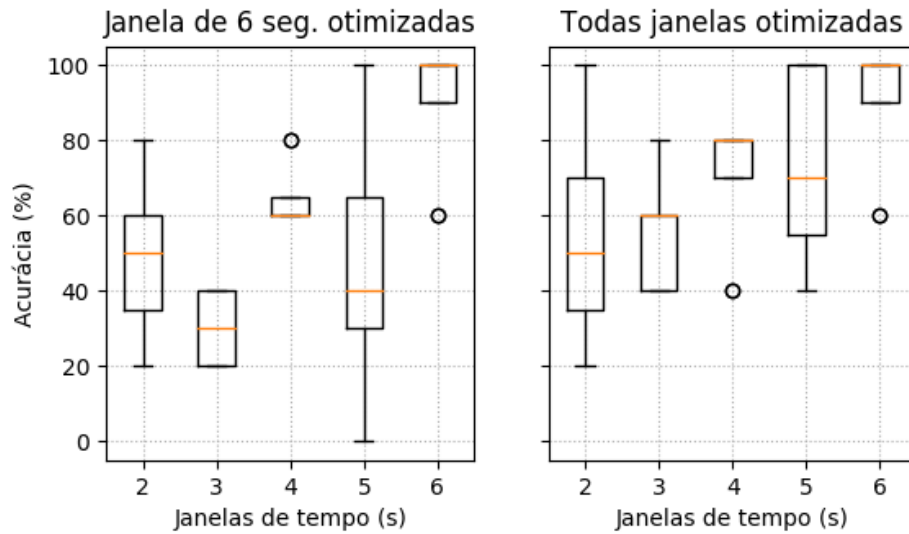


Figura 34 – Distribuição das acurácias obtidas pelo participante 3. No gráfico à esquerda está a distribuição de acurácias otimizando os canais somente da janela de 6 segundos. No gráfico à direita está a distribuição de acurácias otimizando os canais em todas as janelas de tempo.

Tabela 4 – Significância estatística em valor-p para os janelamentos de tempo de cada participante, tomando como base a janela máxima de 6 segundos

	P1	P2	P3	P3 (CAR)	P4
6 seg.	80.00% \pm 15.12%	72.50% \pm 21.21%	90.00% \pm 18.52%	90.00% \pm 18.52%	80.00% \pm 15.12%
6-5 seg.	75.00% \pm 17.73% p = 55.36%	60.00% \pm 18.52% p = 22.98%	45.00% \pm 35.05% p = 0.63%	72.50% \pm 26.05% p = 14.37%	60.00% \pm 15.12% p = 1.92%
6-4 seg.	70.00% \pm 18.52% p = 25.64%	55.00% \pm 17.73% p = 9.50%	65.00% \pm 9.26% p = 0.42%	70.00% \pm 18.52% p = 4.86%	65.00% \pm 27.77% p = 20.11%
6-3 seg.	57.50% \pm 31.05% p = 8.67%	30.00% \pm 15.12% p = 0.04%	30.00% \pm 10.69% p = 0.00%	55.00% \pm 14.14% p = 0.08%	50.00% \pm 18.52% p = 0.32%
6-2 seg.	52.50% \pm 28.16% p = 2.89%	35.00% \pm 23.30% p = 0.46%	47.50% \pm 21.21% p = 0.08%	55.00% \pm 31.62% p = 1.72%	37.50% \pm 24.93% p = 0.10%

pelo participante 3 (sem otimizar os canais) sofreu a queda de desempenho do sistema comparando todas as janelas de tempo. É importante destacar que em contrapartida, os resultados do participante 3 (CAR), manteve o desempenho até a comparação com o janelamento de 4 segundos, fortalecendo assim, a importância de otimizar as combinações de canais como parâmetro do filtro CAR, para os diferentes janelamentos de tempo de uma mesma sessão.

5.2.3 3ª configuração

O objetivo deste experimento foi analisar a diferença da resposta SSVEP das diferentes cores de LED utilizadas no alvo (vermelha, amarela e verde do protótipo de semáforo), para verificar se a cor dos LEDs poderiam influenciar nos resultados apresentados na 1ª e 2ª configuração.

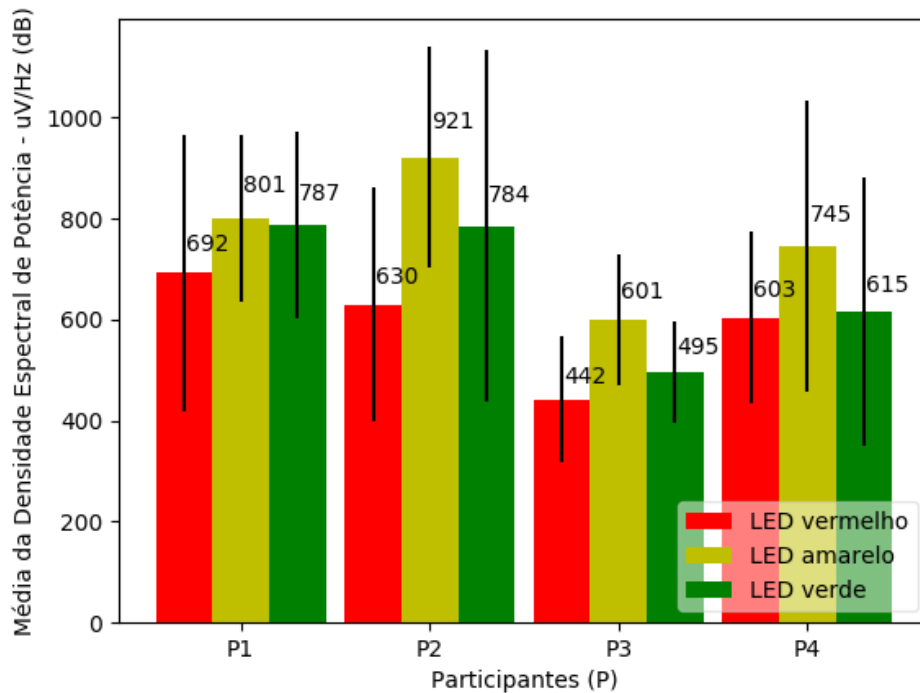


Figura 35 – Gráfico de barras para o valor médio de PSD encontrado em cada sessão, divididos por participante, indicando cada cor de LED.

No trabalho apresentado por Duszyk *et al.* (2014), a diferença das cores sobre os objetos utilizados para estimular o sinal SSVEP, foi realizada por meio da comparação da potência relativa, utilizando um método chamado Perturbação Espectral Relacionada a Eventos (do inglês, *Event Related Spectral Perturbation* - ERSP). Este método pode ser implementado pela ferramenta MNE computando também a densidade espectral de potência em torno da frequência de interesse¹.

Foi desenvolvido o Programa A.9 apresentado no Apêndice A, para computar o PSD dos eventos (luz verde, amarela e vermelha) separadamente das sessões de cada participante. A frequência estimulada foi de 40 Hz para todas as cores de LEDs utilizando somente o canal Oz e então a média e o desvio dos pontos foi calculada com base dos PSD resultantes dos três diferentes estímulos. O gráfico de barras para o valor médio de PSD encontrado em cada sessão, divididos por participante é apresentado na Figura 35.

Os valores apresentados por estes gráficos mostram que a cor do LED amarelo maior que o restante para todos os participantes, porém com um desvio padrão muito grande. Desta forma, a proporção dos valores correspondentes às cores dos LEDs evidenciam que esta seriam um boa característica a ser utilizada pelo classificador, para aumentar o desempenho do sistema. Uma possível resposta a isso, se deve ao fato de que a frequência estimulada em 40 Hz não evocou um sinal SSVEP em torno desta frequência para todas

¹ https://martinos.org/mne/dev/auto_examples/time_frequency/plot_time_frequency_erds.html

as sessões realizadas com os participantes. Dois gráficos de PSD do canal 0z utilizando o método Multi Taper é apresentado na Figura 36, exemplificando esta suposição.

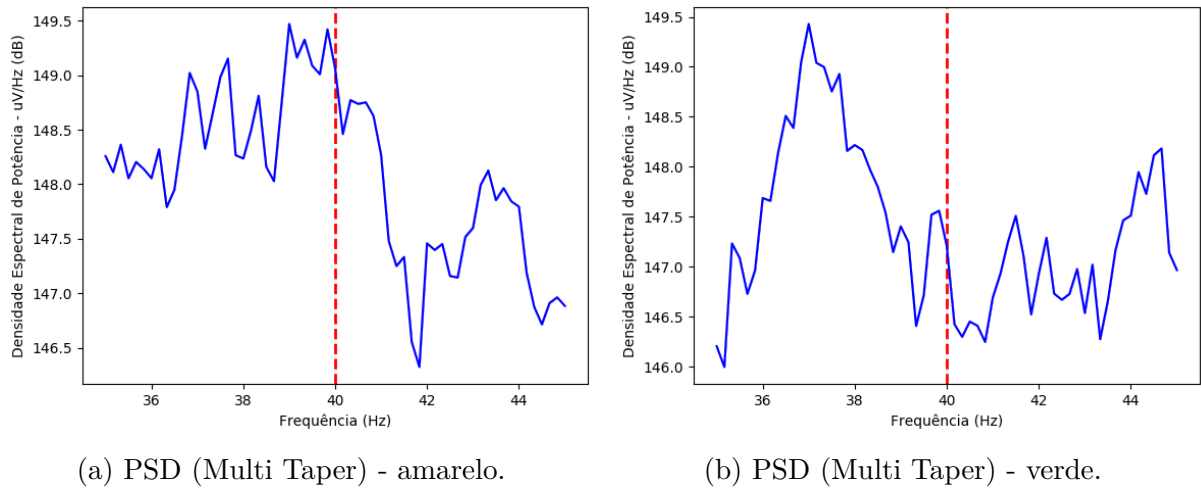


Figura 36 – Gráficos da Densidade Espectral de Potência do canal 0z utilizando o método Multi Taper na frequência evocada em 40 Hz.

Desta forma, foi constatado por estes experimentos, que a característica da intensidade da luz não influencia significativamente no PSD e assim, não contribui para o desempenho do sistema SSVEP-BCI apresentado.

5.3 Discussão sobre a aplicação do sistema SSVEP-BCI em tempo real

Dados os resultados apresentados durante a tese, constamos a possibilidade de desenvolver sistemas SSVEP-BCI para auxiliar na tomada de decisões. Por meio da análise destes resultados no formato *offline*, foi possível detectar algumas falhas, que em sua maior parte, puderam ser contornadas utilizando métodos e técnicas aplicadas no processamento do sinal SSVEP, corrigindo assim as lacunas no desempenho. Estes resultados permitiram também argumentar sobre a construção de um sistema SSVEP-BCI para auxiliar na tomada de decisões em tempo real, no qual esta seção explana.

Uma tomada de decisão no qual foi abordada nesta tese como a “melhor escolha” da ação a ser realizada em um semáforo de trânsito enquanto está dirigindo, ocorreria evidentemente em tempo real. Podemos não só destacar a situação imposta por um semáforo de trânsito, mas generalizar em outras situações, como por exemplo, o controle de uma máquina em uma indústria, no qual esta máquina poderia contar com artifícios para indicar diversas situações que necessitam de atenção por meio de diferentes alvos (no caso LEDs). Tal situação também poderia ser beneficiada por um sistema SSVEP-BCI para auxiliar

em uma decisão, reduzindo a possibilidade do operador da máquina tomar uma decisão incorreta que poderia prejudicá-lo.

Partindo da ideia de aplicar o sistema SSVEP-BCI em uma situação real, os métodos desenvolvidos para o processamento do sinal SSVEP deveriam sofrer algumas modificações para a execução de uma avaliação *online*. Em contraste com os experimentos *offline* apresentados por esta tese, em que todo o conjunto de dados deve estar disponível, na configuração de um experimento *online*, é necessário processar os dados de entrada em pequenos blocos, que equivalem a um curto prazo de tempo. Normalmente, os fragmentos de dados analisados em tempo real, possuem um tamanho de apenas algumas amostras (ou blocos), mesmo que o equipamento de EEG tenha uma alta taxa de transferência. O processamento *online* de um sistema SSVEP-BCI, também é dividido em duas etapas: (1) treinamento, que não iria mudar em relação ao que já foi avaliado nesta tese e (2) teste, que ocorre em tempo de execução durante o experimento. Isso leva a diferenças em algumas das etapas de processamento do sinal SSVEP:

- Ao utilizar os filtros temporais nos dados, segundo Venthur *et al.* (2015), sugere-se utilizar um método de filtro temporal com valores de dados em atraso, compondo a quantidade mínima necessária para que os filtros temporais (*notch* e passagem de banda) possam ser aplicados e assim, utilizar os mesmos dados como se estivesse filtrando todo o conjunto de dados de uma só vez;
- Filtros temporais poderiam ser otimizados somente na etapa de treinamento, no qual a configuração encontrada seria “copiada” para o teste.

As demais etapas de processamento podem ocorrer da mesma forma que foram realizadas nos experimentos *offline*, pois a maior parte do custo de tempo, ocorre para extrair e selecionar características e estimar o classificador de dados, compreende na etapa de treinamento e apenas seria aplicada no teste *online*. Para se ter uma ideia da modelagem de um sistema *online* SSVEP-BCI para auxiliar na tomada de decisões, um fluxograma que compreende todos os passos deste sistema é apresentado na Figura 37.

Neste fluxograma, generalizamos a questão da aquisição dos dados EEG, no sentido de que seria o mesmo processo tanto para o treinamento quanto para o teste. O treinamento poderia seguir os mesmo padrões das análises já realizadas por esta tese, gerando ao final um modelo de classificação (estimador). Já o teste, utilizará recursos menos custosos, principalmente no pré-processamento dos dados, e no final de seu processo, utiliza o estimador treinado para classificar os dados e apresentá-los (ou realizar alguma ação). Além disso, o processo de aquisição e teste seria realizado continuamente conforme a necessidade da aplicação, pois os dados fornecidos pela aquisição ocorrem em pequenos blocos, em tempo real.

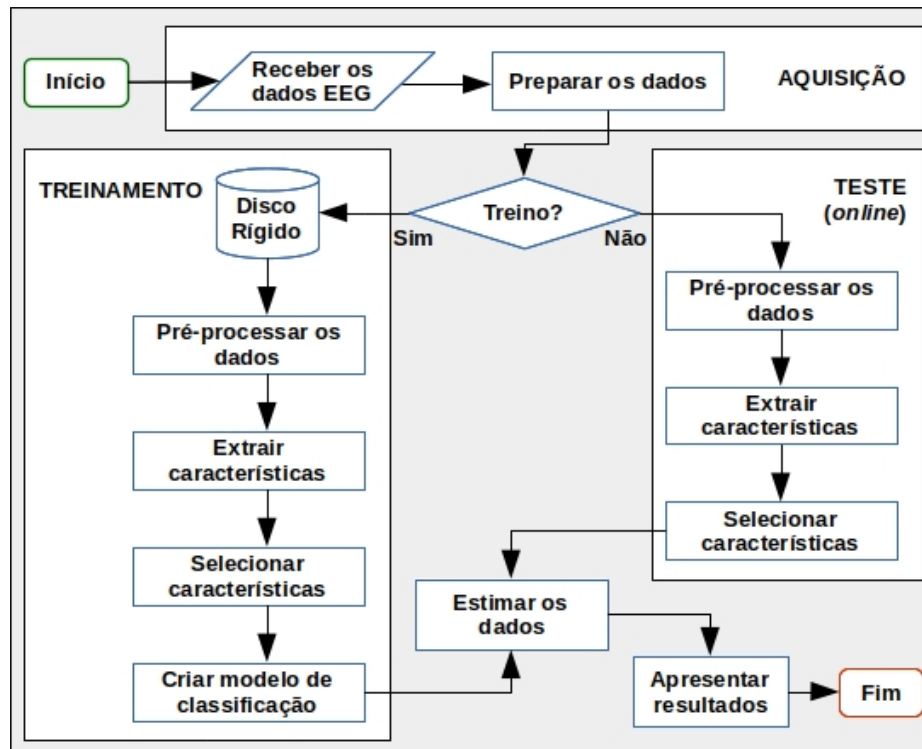


Figura 37 – Fluxograma exemplificando a modelagem de um sistema *online* SSVEP-BCI para auxiliar na tomada de decisões.

Uma questão pertinente pode ser levantada diante da questão “análise *offline* versus análise *online*”: O quanto vale a pena otimizar parâmetros dos métodos de processamento do sinal SSVEP, para diversas janelas de tempo, quando a intenção seria aplicar o sistema em tempo real? As maiores características de um sistema para auxiliar na tomada de decisão, é agir de forma rápida e que preferencialmente tivesse o tempo de treinamento curto. Em um sistema real para a tomada de decisão, não iríamos fixar uma janela de tempo específica. Seria necessário encontrar um limiar em termos de acurácia e permitir que o classificador forneça a resposta à ação assim que atingir este limiar.

5.4 Considerações finais

Neste capítulo foi apresentado uma análise *offline* de um sistema SSVEP-BCI com a utilização de baixas frequências (8, 10 e 12 Hz) e altas frequências consideradas “não visíveis” aos participantes (37, 40 e 43 Hz). Os resultados foram bons, considerando que a média da acurácia para a maior janela de tempo, acima de 80%. Diante destes resultados, consideramos viável a construção de um sistema SSVEP-BCI para auxiliar na tomada de decisões *online*, como apresentado na seção anterior, porém levando em consideração o mesmo cenário controlado apresentado. O próximo capítulo apresenta as conclusões e alguns trabalhos futuros.

6 Conclusões e trabalhos futuros

Nesta tese de doutorado foi desenvolvido diversos experimentos baseados em SSVEP-BCI, lidando com baixas e altas frequências de cintilação para viabilizar um sistema para auxiliar na tomada de decisões. Tivemos como resultado da avaliação *offline* em experimentos com Interfaces Cérebro-Computador (BCI), utilizando o paradigma de Potenciais Evocados Visualmente em Regime Estacionário (SSVEP) em baixas e altas frequências de cintilação. Os experimentos foram avaliados com quatro participantes utilizando um protótipo de semáforos de trânsito construído com LEDs, para simular uma situação em que um indivíduo deve tomar uma decisão. Os seguintes produtos resultantes do desenvolvimento da tese foram gerados: (1) códigos para análise e avaliação de sistemas SSVEP-BCI, (2) protótipos de semáforo de trânsito para estimular o sinal SSVEP em diferentes frequências de cintilação e (3) um capacete para acoplar o hardware responsável pela aquisição dos dados de eletroencefalografia (EEG) e para acoplar os eletrodos de prata-cloreto de prata (Ag-AgCl) de forma que se ajustem adequadamente sem causar danos ou incômodo aos participantes.

Concluimos por meio dos resultados apresentados no Capítulo 5, que é possível construir um sistema SSVEP-BCI para auxiliar na tomada de decisões em tempo real (*online*) e até mesmo em situações reais no qual seja possível mapear estímulos SSVEP para diferentes possibilidades da tomada de decisão. Foi levantado também questões pertinentes em relação ao que deve ser feito para adaptar as avaliações realizadas para construção de um sistema no formato *online*.

Além do objetivo principal alcançado, levantamos algumas contribuições científicas e técnicas, resultante do andamento de todo o trabalho realizado. Listamos a seguir tais contribuições.

- A primeira contribuição foi realizar uma avaliação dos estímulos SSVEP em frequências de cintilação acima das tradicionalmente utilizadas. Os trabalhos que utilizam altas frequências (acima de 30 Hz), não fornecem informações suficientes para a análise do sinal SSVEP, para que o experimento pudesse ser reproduzido;
- Uma avaliação, mesmo com resultados não satisfatórios, para as diferentes intensidades de luz, estimuladas na mesma frequência por meio da análise do PSD, concluindo que esta não é uma alternativa viável para este tipo de análise;
- Um algoritmo para estimar as melhores combinações de canais para serem utilizados como parâmetro do filtro temporal CAR. Mesmo que o algoritmo não seja o provado

como ótimo, foi possível obter uma boa estimativa e melhorar o desempenho de forma significativa em algumas situações;

- O desenvolvimento de um cenário completo para experimentos SSVEP-BCI, contemplando todos os componentes físicos e lógicos do sistema; e, não obstante
- O fornecimento de exemplos realizados para o processamento de sinais SSVEP aos mantenedores da ferramenta MNE, pois tal comunidade ainda não disponibilizou exemplos pertinentes a este paradigma de BCI.

Trabalhos e perspectivas futuras

Dados os resultados obtidos nesta tese, além de diversos fatores observados durante o desenvolvimento dos experimentos, há uma série de trabalhos futuros possíveis de se realizar, e listamos a seguir alguns deles.

Primeiramente, é apontado como um trabalho futuro, a avaliação completa deste sistema no formato *online*, criando uma situação real em que o indivíduo tenha que tomar uma decisão e como resultado do auxílio fornecido pelo sistema, um *feedback* seja fornecido a ele.

A respeito dos métodos de processamento do sinal SSVEP:

- O filtro temporal utilizando o método CAR possui melhores respostas com o conjunto certo de canais escolhidos como referência. O problema é que o método utilizado para encontrar a melhor combinação de canais foi basicamente manual, como foi explicado na Seção 5.1 e, além disso, o método utilizado não contempla todas as combinações dos canais EEG disponíveis. Um trabalho futuro é desenvolver um algoritmo que possa encontrar a melhor combinação de canais para serem utilizados no filtro temporal CAR. Na prática, este problema possui no pior caso, testar todas as combinações possíveis que podem ser realizadas. Supondo que existam 8 canais disponíveis, é possível ter 255 combinações diferentes e a execução da análise com uma combinação de canais demora em média 13 segundos. No pior caso (se não for encontrado uma combinação que dê 100% de acurácia), o problema pode demorar aproximadamente 1 hora;
- A seleção de características por meio do método da Eliminação Recursiva de Características (RFE), pode ser aplicada somente se o classificador SVM utilizar o *kernel* linear. Um trabalho futuro é comparar outros métodos de seleção de características para utilizar outros *kernels* disponíveis no classificador SVM e, além disso, será possível otimizar o parâmetro *gamma*, no qual não está disponível no *kernel* linear;

- Ainda é necessário realizar uma comparação entre os métodos de classificação, tais como o LDA, KNN e outros que podem ser utilizados neste tipo de aplicação BCI, que não foram citados no trabalho;
- Uma avaliação do sistema *online* ainda deve ser realizada, de modo que o sistema SSVEP-BCI obtido possa ser aplicado em uma situação real.

A respeito do método para evocação do sinal SSVEP e do equipamento EEG:

- A taxa de transferência do equipamento de EEG utilizado nesta tese é de 250 Hz. Seria possível aumentar esta taxa de transferência utilizando um *shield* no OpenBCI board que transformaria o método de comunicação do equipamento com o computador de Bluetooth para Wi-Fi, aumentando a taxa de transferência para 1 kHz. Desta forma, o trabalho futuro seria refazer os experimentos com este novo equipamento para obter valores de PSD mais confiáveis e consequentemente melhores resultados;
- O capacete utilizado (Ultracortex Mark III) para acoplar os eletrodos e posicioná-los sobre a cabeça do participante, possui não permite posicionar os eletrodos em alguns locais importantes para a obtenção do sinal SSVEP. Outros modelos de capacete podem ser testados para melhorar a configuração dos eletrodos. Em contrapartida, para a maior parte dos resultados, poucos eletrodos foram considerados para uma boa resposta VEP. Um trabalho futuro é modelar um capacete (ou arco) mais fácil de tirar e colocar na cabeça do indivíduo.
- Muitas interferências ocorreram nos experimentos utilizando o protótipo de semáforo, para evocar o sinal SSVEP utilizando LEDs. Para um trabalho futuro, sugerimos isolar os equipamentos externos e descobrir as fontes de interferência, para que os resultados não sofram alterações.

Referências

- BAYLISS, J. D.; BALLARD, D. H. Single trial P3 epoch recognition in a virtual environment. *Neurocomputing*, v. 32–33, p. 637 – 642, 2000. ISSN 0925-2312. Citado na página 41.
- BRONZINO, J.; PETERSON, D. *The Biomedical Engineering Handbook*. Boca Raton, FL: CRC Press LLC, 2015. v. 4. Citado na página 24.
- CAO, T.; WAN, F.; MAK, P. U.; MAK, P. I.; VAI, M. I.; HU, Y. Flashing color on the performance of ssvep-based brain-computer interfaces. In: *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. San Diego, CA, USA: IEEE, 2012. p. 1819–1822. ISSN 1094-687X. Citado na página 63.
- CARVALHO, S. N.; COSTA, T. B.; URIBE, L. F.; SORIANO, D. C.; YARED, G. F.; CORADINE, L. C.; ATTUX, R. Comparative analysis of strategies for feature extraction and classification in SSVEP BCIs. *Biomedical Signal Processing and Control*, v. 21, p. 34 – 42, 2015. ISSN 1746-8094. Citado 2 vezes nas páginas 29 e 31.
- CHABUDA, A.; DURKA, P.; ŻYGIEREWICZ, J. High frequency ssvep-bci with hardware stimuli control and phase-synchronized comb filter. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, v. 26, n. 2, p. 344–352, Feb 2018. ISSN 1534-4320. Citado na página 18.
- CHAUDHARY, U.; BIRBAUMER, N.; RAMOS-MURGUIALDAY, A. Brain-computer interfaces for communication and rehabilitation. Nature Publishing Group, a division of Macmillan Publishers Limited. All Rights Reserved, Cham, p. 513–525, 2016. Citado na página 17.
- CHEN, X.; WANG, Y.; ZHANG, S.; GAO, S.; HU, Y.; GAO, X. A novel stimulation method for multi-class SSVEP-BCI using intermodulation frequencies. *J Neural Eng*, v. 14, n. 2, p. 026013, Apr 2017. Citado 2 vezes nas páginas 27 e 38.
- CHIAVENATO, I. *Introdução a Teoria Geral Da Administração*. 9. ed. Brasil: Elsevier Brasil, 2014. 678 p. ISBN 9788520436691. Citado na página 34.
- CHUMERIN, N.; MANYAKOV, N. V.; VLIET, M. van; ROBBEN, A.; COMBAZ, A.; HULLE, M. M. V. Steady-state visual evoked potential-based computer gaming on a consumer-grade eeg device. *IEEE Transactions on Computational Intelligence and AI in Games*, v. 5, n. 2, p. 100–110, June 2013. ISSN 1943-068X. Citado na página 26.
- DUSZYK, A.; BIERZYŃSKA, M.; RADZIKOWSKA, Z.; MILANOWSKI, P.; KUŚ, R.; SUFFCZYŃSKI, P.; MICHALSKA, M.; LABECKI, M.; ZWOLIŃSKI, P.; DURKA, P. Towards an optimization of stimulus parameters for brain-computer interfaces based on steady state visual evoked potentials. *PLoS ONE*, v. 9, n. 11, p. e112099, 2014. Citado 2 vezes nas páginas 38 e 77.
- FALLER, J.; SCHERER, R.; FRIEDRICH, E. V.; COSTA, U.; OPISSO, E.; MEDINA, J.; MULLER-PUTZ, G. R. Non-motor tasks improve adaptive brain-computer interface

performance in users with severe motor impairment. *Front Neurosci*, v. 8, p. 320, 2014. Citado na página 28.

FAZEL-REZAI, R.; AHMAD, W. *P300-based Brain-Computer Interface paradigm design*. USA: INTECH Open Access Publisher, 2011. Citado na página 27.

FOUAD, M. M.; AMIN, K. M.; EL-BENDARY, N.; HASSANIEN, A. E. Brain computer interface: A review. In: HASSANIEN, A. E.; AZAR, A. T. (Ed.). *Brain-Computer Interfaces: Current Trends and Applications*. Cham: Springer International Publishing, 2015. p. 3–30. ISBN 978-3-319-10978-7. Citado na página 17.

GRAIMANN, B.; ALLISON, B.; PFURTSCHELLER, G. Brain-computer interfaces: A gentle introduction. In: GRAIMANN, B.; PFURTSCHELLER, G.; ALLISON, B. (Ed.). *Brain-Computer Interfaces*. USA: Springer Berlin Heidelberg, 2010, (The Frontiers Collection). p. 1–27. ISBN 978-3-642-02090-2. Citado 2 vezes nas páginas 25 e 26.

GRAMFORT, A.; LUESSI, M.; LARSON, E.; ENGEMANN, D.; STROHMEIER, D.; BRODBECK, C.; GOJ, R.; JAS, M.; BROOKS, T.; PARKKONEN, L.; HäMÄLÄINEN, M. Meg and eeg data analysis with mne-python. *Frontiers in Neuroscience*, v. 7, p. 267, 2013. ISSN 1662-453X. Disponível em: <<https://www.frontiersin.org/article/10.3389/fnins.2013.00267>>. Citado na página 43.

HALDER, S.; PINEGGER, A.; KÄTHNER, I.; WRIESSNEGGER, S. C.; FALLER, J.; ANTUNES, J. B. P.; MÜLLER-PUTZ, G. R.; KÜBLER, A. Brain-controlled applications using dynamic {P300} speller matrices. *Artificial Intelligence in Medicine*, v. 63, n. 1, p. 7 – 17, 2015. ISSN 0933-3657. Citado na página 27.

HARDING, G.; WILKINS, A. J.; ERBA, G.; BARKLEY, G. L.; FISHER, R. S. Photic- and pattern-induced seizures: expert consensus of the Epilepsy Foundation of America Working Group. *Epilepsia*, v. 46, n. 9, p. 1423–1425, Sep 2005. Citado na página 19.

HWANG, C.; CHEN, M.; SHIH, C.; CHEN, H.; LIU, W. K. Apply scikit-learn in python to analyze driver behavior based on obd data. In: *2018 32nd International Conference on Advanced Information Networking and Applications Workshops (WAINA)*. Krakow, Poland: IEEE, 2018. p. 636–639. Citado na página 31.

HWANG, H.-J.; LIM, J.-H.; JUNG, Y.-J.; CHOI, H.; LEE, S. W.; IM, C.-H. Development of an ssvep-based BCI spelling system adopting a qwerty-style LED keyboard. *Journal of Neuroscience Methods*, v. 208, n. 1, p. 59 – 65, 2012. ISSN 0165-0270. Citado na página 36.

JURCAK, V.; TSUZUKI, D.; DAN, I. 10/20, 10/10, and 10/5 systems revisited: Their validity as relative head-surface-based positioning systems. *NeuroImage*, v. 34, n. 4, p. 1600 – 1611, 2007. ISSN 1053-8119. Citado na página 24.

KHALILIARDALI, Z.; LOZANO, R. C.; ZHANG, H.; GHEORGHE, L. A.; MILLÁN, J. d. R. Single Trial Classification of Neural Correlates of Anticipatory Behavior during Real Car Driving. In: *Proceedings of the 6th International Brain-Computer Interface Meeting*. Asilomar, California, USA: EPFL, 2016. Citado na página 40.

LIN, C. T.; KO, L. W.; LIN, K. L.; LIANG, S. F.; KUO, B. C.; CHUNG, I. F.; VAN, L. D. Classification of driver's cognitive responses using nonparametric single-trial eeg

- analysis. In: *2007 IEEE International Symposium on Circuits and Systems*. New Orleans, LA, USA: IEEE, 2007. p. 2019–2023. ISSN 0271-4302. Citado na página 42.
- LIN, K.; CINETTO, A.; WANG, Y.; CHEN, X.; GAO, S.; GAO, X. An online hybrid bci system based on ssvep and emg. *Journal of Neural Engineering*, v. 13, n. 2, p. 026020, 2016. Citado na página 17.
- LIN, Y.-P.; WANG, Y.; JUNG, T.-P. Assessing the feasibility of online ssvep decoding in human walking using a consumer eeg headset. *Journal of NeuroEngineering and Rehabilitation*, v. 11, n. 1, p. 119, 2014. ISSN 1743-0003. Citado na página 17.
- MARACINE, M.; RADU, A.; CIOBANU, V.; POPESCU, N. Brain computer interface architectures and classification approaches. In: *2017 21st International Conference on Control Systems and Computer Science (CSCS)*. Bucharest, Romania: IEEE, 2017. p. 217–222. ISSN 2379-0482. Citado na página 31.
- MARSHALL, D.; COYLE, D.; WILSON, S.; CALLAGHAN, M. Games, gameplay, and bci: The state of the art. *IEEE Transactions on Computational Intelligence and AI in Games*, v. 5, n. 2, p. 82–99, 2013. Cited By 24. Citado na página 36.
- MARTIŠIUS, I.; DAMAŠEVIČIUS, R. A prototype ssvep based real time bci gaming system. *Intell. Neuroscience*, Hindawi Publishing Corp., New York, NY, United States, v. 2016, p. 18–, mar. 2016. ISSN 1687-5265. Citado 2 vezes nas páginas 17 e 31.
- MCCOY, E. J.; WALDEN, A. T.; PERCIVAL, D. B. Multitaper spectral estimation of power law processes. *IEEE Transactions on Signal Processing*, v. 46, n. 3, p. 655–668, Mar 1998. ISSN 1053-587X. Citado na página 29.
- MCFARLAND, D. J.; MCCANE, L. M.; DAVID, S. V.; WOLPAW, J. R. Spatial filter selection for eeg-based communication. *Electroencephalography and Clinical Neurophysiology*, v. 103, n. 3, p. 386 – 394, 1997. ISSN 0013-4694. Citado na página 29.
- MISTRY, K. S.; PELAYO, P.; ANIL, D. G.; GEORGE, K. An ssvep based brain computer interface system to control electric wheelchairs. In: *2018 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*. Houston, TX, USA: IEEE, 2018. p. 1–6. Citado na página 17.
- MOLINA, G. G.; IBÁÑEZ, D.; MIHAJLOVIĆ, V.; CHESTAKOV, D. Detection of high frequency steady state visual evoked potentials for brain-computer interfaces. In: *2009 17th European Signal Processing Conference*. Glasgow, UK: IEEE, 2009. p. 646–650. Citado na página 19.
- Mühl, C.; Gürkök, H.; Plass-Oude Bos, D.; Thurlings, M. E.; Scherffig, L.; Duvinage, M.; Elbakyan, A. A.; Kang, S.; Poel, M.; Heylen, D. Bacteria hunt: Evaluating multi-paradigm bci interaction. *Journal on Multimodal User Interfaces*, Springer Verlag, Berlin, v. 4, n. 1, p. 11–25, March 2010. Open Access. Citado na página 25.
- NEUPER, C.; SCHERER, R.; REINER, M.; PFURTSCHELLER, G. Imagery of motor actions: Differential effects of kinesthetic and visual-motor mode of imagery in single-trial {EEG}. *Cognitive Brain Research*, v. 25, n. 3, p. 668 – 677, 2005. ISSN 0926-6410. Citado na página 28.

- NIEDERMEYER, E.; SILVA, F. L. da. *Electroencephalography: Basic Principles, Clinical Applications, and Related Fields*. 5th. ed. USA: Lippincott Williams & Wilkins, 2004. Hardcover. ISBN 0781751268. Citado na página 24.
- OU, G.; WANG, Y.; PANG, W.; COGHILL, G. M. Large margin distribution machine recursive feature elimination. *IEEE*, Hangzhou, China, p. 1518–1523, Nov 2017. Citado na página 31.
- PERAKAKIS, M.; POTAMIANOS, A. An affective evaluation tool using brain signals. In: *Proceedings of the Companion Publication of the 2013 International Conference on Intelligent User Interfaces Companion*. New York, NY, USA: ACM, 2013. (IUI '13 Companion), p. 105–106. ISBN 978-1-4503-1966-9. Disponível em: <<http://doi.acm.org/10.1145/2451176.2451222>>. Citado na página 36.
- PFURTSCHELLER, G.; NEUPER, C. Motor imagery direct communication. *Proceedings of the IEEE*, v. 89, n. 7, p. 1123–1134, 2001. Cited By 0. Citado na página 28.
- PRASHANT, P.; JOSHI, A.; GANDHI, V. Brain computer interface: A review. In: *2015 5th Nirma University International Conference on Engineering (NUICONE)*. Ahmedabad, India: IEEE, 2015. p. 1–6. Citado na página 17.
- REGAN, D. Steady-state evoked potentials. *J. Opt. Soc. Am.*, OSA, v. 67, n. 11, p. 1475–1489, Nov 1977. Citado na página 26.
- RESALAT, S.; SABA, V. A practical method for driver sleepiness detection by processing the eeg signals stimulated with external flickering light. *Signal, Image and Video Processing*, Springer London, v. 9, n. 8, p. 1751–1757, 2015. ISSN 1863-1703. Citado na página 40.
- SAKURADA, T.; KAWASE, T.; KOMATSU, T.; KANSAKU, K. Use of high-frequency visual stimuli above the critical flicker frequency in a ssvep-based bmi. *Clinical Neurophysiology*, v. 126, n. 10, p. 1972 – 1978, 2015. ISSN 1388-2457. Citado 4 vezes nas páginas 18, 39, 54 e 72.
- SANEI, S.; CHAMBERS, J. A. *EEG Signal Processing*. USA: Wiley-Interscience, 2007. Hardcover. ISBN 0470025816. Citado na página 25.
- SCHUH, A. R.; LIMA, A.; MORCHE, G.; MOSSMANN, J. a.; BEZ, M. R. Control of a wheelchair simulator in a three-dimensional environment using eye blink detection through non-invasive brain-computer interface. In: *Proceedings of the XV International Conference on Human Computer Interaction*. New York, NY, USA: ACM, 2014. (Interacción '14), p. 76:1–76:2. ISBN 978-1-4503-2880-7. Citado na página 36.
- SHENOI, B. A. *Introduction to Digital Signal Processing and Filter Design*. USA: Wiley-Interscience, 2005. ISBN 0471464821. Citado na página 29.
- SMOLA, A. J.; SCHÖLKOPF, B. *A tutorial on support vector regression*. 2004. Citado na página 32.
- THAKOR, N. *Biopotentials and Electrophysiology Measurement: Measurements, instrumentations and sensors handbook*. Boca Raton, FL: CRC Press LLC, 1999. Citado na página 23.

VENTHUR, B.; DÄHNE, S.; HÖHNE, J.; HELLER, H.; BLANKERTZ, B. Wyrn: A brain-computer interface toolbox in python. *Neuroinformatics*, v. 13, n. 4, p. 471–486, Oct 2015. ISSN 1559-0089. Disponível em: <<https://doi.org/10.1007/s12021-015-9271-8>>. Citado na página 79.

VILIC, A. *AVI SSVEP Dataset*. 2014. Disponível em: <<http://www.setzner.com/avi-ssvep-dataset>>. Acesso em: 19/04/2017. Citado 4 vezes nas páginas 8, 44, 45 e 51.

VILIC, A.; KJAER, T. W.; THOMSEN, C. E.; PUTHUSSERYPADY, S.; SORENSEN, H. B. D. Dtu bci speller: An ssvep-based spelling system with dictionary support. In: *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. Osaka, Japan: IEEE, 2013. p. 2212–2215. ISSN 1094-687X. Citado na página 44.

WEBSTER, J. G. *Medical Instrumentation Application and Design*. 4th. ed. USA: Wiley, 2009. Citado na página 24.

YANG, B. hua; YAN, G. zheng; WU, T.; YAN, R. guo. Subject-based feature extraction using fuzzy wavelet packet in brain-computer interfaces. *Signal Processing*, v. 87, n. 7, p. 1569 – 1574, 2007. ISSN 0165-1684. Citado na página 29.

ZHANG, C.; LIU, C.; ZHANG, X.; ALMPANIDIS, G. An up-to-date comparison of state-of-the-art classification algorithms. *Expert Systems with Applications*, v. 82, p. 128 – 150, 2017. ISSN 0957-4174. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0957417417302397>>. Citado na página 31.

ZHANG, N.; YU, Y.; YIN, E.; ZHOU, Z. Performance of virtual stimulus motion based on the ssvep-bci. In: *2016 International Symposium on Computer, Consumer and Control (IS3C)*. Xi'an, China: IEEE, 2016. p. 656–659. Citado na página 37.

ZHU, D.; BIEGER, J.; MOLINA, G. G.; AARTS, R. M. A survey of stimulation methods used in ssvep-based bcis. *Intell. Neuroscience*, Hindawi Publishing Corp., New York, NY, United States, v. 2010, p. 1:1–1:12, jan. 2010. ISSN 1687-5265. Citado 2 vezes nas páginas 17 e 26.

APÊNDICE A – Programas desenvolvidos

A.1 Código da avaliação do 1º experimento com os dados *single target*

```

1  #!/ coding: utf-8
2
3  from bci_utils import view_psd_multitaper, view_psd
4  from mne import EpochsArray
5  from mne.decoding import PSDEstimator
6  from mne.io.meas_info import create_info
7  from sklearn.feature_selection import RFE
8  from sklearn.preprocessing import StandardScaler
9  from sklearn import svm
10 import numpy as np
11 import scipy.io
12 import sys
13
14 reload(sys)
15 sys.setdefaultencoding('utf8')
16
17 class Model:
18
19     def __init__(self, data, target, labels):
20         self.data = data
21         self.target = target
22         self.labels = labels
23
24     def training(self, estimator):
25         # utilização da Eliminação de Características Recursiva (RFE) como um
26         # selector de características
27         self.classifier = RFE(estimator=estimator)
28         self.classifier.fit(self.data, self.labels)
29
30 def avi_base(fname, test_part):
31     """Função para criar objetos MNE epochs, retirado dos dados EEG dos
32     arquivos Matlab."""
33
34     # carregamento dos dados do arquivo matlab
35     data = scipy.io.loadmat(fname)['Data']
36     # construindo "event_id" e "events" p/ construção do objeto MNE
37     t_freqs = ['%s' % str(t) for t in data['TargetFrequency'][0][0][0]]
38     # obtendo valores dos alvos das sessões

```

```

39     f_freqs = set(t_freqs)
40     event_id = dict(zip(f_freqs, range(1, len(f_freqs) + 1)))
41     # dividindo os eventos em 'treino' e 'teste'
42     events_tr, events_te = list(), list()
43     for i, t in enumerate(t_freqs):
44         if i % 3 == test_part:
45             events_te.append([i, 0, event_id[t]])
46         else:
47             events_tr.append([i, 0, event_id[t]])
48     events_tr, events_te = np.array(events_tr), np.array(events_te)
49     # configurando a frequência de amostragem do amplificador
50     sfreq = data['AmpSamplingFrequency'][0][0][0][0]
51     # nome dos canais EEG
52     ch_names = [data['ElectrodePlacement'][0][0][0]]
53     # tipo dos canais EEG
54     ch_types = ['eeg']
55     # construção da descrição 'info' para o objeto MNE
56     info = create_info(ch_names, sfreq, ch_types)
57     # construção do vetor 'MNE epochs' dividindo entre dados de treino e teste
58     data = data['EEG'][0][0]
59     data = np.expand_dims(data.T, axis=1)
60     data_tr, data_te = list(), list()
61     for i, d in enumerate(data):
62         if i % 3 == test_part:
63             data_te.append(d)
64         else:
65             data_tr.append(d)
66     data_tr, data_te = np.array(data_tr), np.array(data_te)
67     epochs_tr = EpochsArray(data_tr, info, events=events_tr, event_id=event_id,
68                             verbose=False)
69     # construção do vetor final
70     epochs_te = EpochsArray(data_te, info, events=events_te, event_id=event_id,
71                             verbose=False)
72     return epochs_tr, epochs_te
73
74
75 def feature_extraction(*base_list):
76     # configurando o desvio padrão
77     sd = 0.3
78     list_features = list()
79     for base in base_list:
80         sfreq = base.info['sfreq']
81         # transformando dois vetores 1-dimensionais em um só (único canal)
82         epochs_data = np.squeeze(base.get_data(), axis=1)
83         features = list()
84         minor_shape = list()
85         for target in base.event_id:

```

```

86         y = list()
87         for i in range(len(base)):
88             y.append(int(target == base[i].event_id.keys()[0]))
89             # adequando a variação entre a frequência mínima e máxima
90             fmin = float(target) - sd
91             fmax = float(target) + sd
92             psd = PSDEstimator(sfreq=sfreq, fmin=fmin, fmax=fmax)
93             X = psd.transform(epochs_data)
94             # normalização do formato dos vetores
95             minor_shape.append(X.shape[1])
96             features.append(Model(X, target, y))
97         for f in features:
98             # configurando para o máximo da menor forma normalizada
99             f.data = f.data[:, :min(minor_shape)]
100     list_features.append(features)
101     return list_features
102
103 def main(subject=2, test_part=0):
104     path = 'AVI_SSVEP_DATASET/single/'
105     data_tr, data_te = avi_base('%sSub%s_singletarget.mat' % (path, subject),
106                                test_part=test_part)
107     features_tr, features_te = feature_extraction(data_tr, data_te)
108     # aplicando o pre-processamento usando uniformização dos dados nas
109     # características extraídas. Este método usa a padronização das
110     # características pela remoção da média e escalando p/ variancia única
111     ss = StandardScaler()
112     for model in features_tr:
113         model.data = ss.fit_transform(model.data)
114     for model in features_te:
115         model.data = ss.transform(model.data)
116     # treinando o classificador SVM - Cada participante realizou 2 sessões.
117     # 2ª sessão possui maior diversidade de classes
118     # As duas sessões possum 10 exemplos
119     estimator = svm.SVC(kernel='linear', C=2) # utilizando SVM como estimador
120     for model in features_tr:
121         model.training(estimator)
122     # teste utilizando a 1ª sessão (menor variabilidade de classes)
123     n_t = len(features_te[0].labels) # número de testes
124     # reformatando vetor para aplicação dos testes
125     tests = np.array([[m.data[e] for m in features_te] for e in range(n_t)])
126     # aplicando o classificador e realizando o voto
127     votes = list()
128     labels = [f.target for f in features_te]
129     for test in tests:
130         vote_set = list()
131         for model in features_tr:
132             vote_set.append(model.classifier.predict(test))

```

```

133         # obtendo o índice mais votado
134         votes.append(labels[sum(vote_set).argmax()])
135         # mostrando os resultados
136         template = [data_te[i].event_id.keys()[0] for i in range(len(data_te))]
137         score = 0
138         for t, v in zip(template, votes):
139             score += 1 if t == v else 0
140         print '%.1f %%' % (score * 100 / len(template))
141         print template
142         print votes
143         # visualização da densidade espectral de potência (PSD)
144         for i in range(len(data_te)):
145             view_psd_multitaper(data_te[i], 4, 14)
146
147 if __name__ == '__main__':
148     from sys import argv
149     if len(argv) > 1:
150         try:
151             subject = int(argv[1])
152             test = int(argv[2])
153         except ValueError:
154             print 'Parâmetros deve ser Int. (%s).' % type(argv[1])
155         else:
156             if 1 <= subject <= 4:
157                 main(subject, test)
158             else:
159                 print 'Parâmetro deve estar entre 1 e 4.'
160     else:
161         main()

```

A.2 Código da avaliação do 1º experimento *multi target*

```

1 # ... cabeçalho igual ao 1º experimento single target
2
3 class Model:
4     # ... classe igual ao 1º experimento single target
5
6     def avi_base(fname):
7         """Função para criar objetos MNE epochs, retirado dos dados EEG dos
8         arquivos Matlab."""
9
10        # carregamento dos dados do arquivo matlab
11        data = scipy.io.loadmat(fname)['Data']
12        # construindo "event_id" e "events" p/ construção do objeto MNE
13        t_freqs = ['%s' % str(t) for t in data['TargetFrequency'][0][0][0]]
14        f_freqs = data['FlickeringFrequencies'][0][0].flatten()

```



```

15     f_freqs = ['%s' % str(f) for f in f_freqs]
16     # removendo os alvos não utilizados nas sessões
17     f_freqs = set.intersection(set(t_freqs), set(f_freqs))
18     event_id = dict(zip(f_freqs, range(1, len(f_freqs) + 1)))
19     events = [[i, 0, event_id[t]] for i, t in enumerate(t_freqs)]
20     events = np.array(events)
21     # configurando a frequência de amostragem do amplificador
22     sfreq = data['AmpSamlingFrequency'][0][0][0][0]
23     # nome dos canais EEG
24     ch_names = [data['ElectrodePlacement'][0][0][0]]
25     # tipo dos canais EEG
26     ch_types = ['eeg']
27     # construção da descrição 'info' para o objeto MNE
28     info = create_info(ch_names, sfreq, ch_types)
29     # construção do vetor 'MNE epochs'
30     data = data['EEG'][0][0]
31     data = np.expand_dims(data.T, axis=1)
32     epochs = EpochsArray(data, info, events=events, event_id=event_id,
33                           verbose=False)
34     return epochs
35
36 def feature_extraction(*base_list):
37     # ... função igual ao 1º experimento single target
38
39 def main(subject, swap):
40     path = 'AVI_SSVEP_DATASET/multi/'
41     if swap:
42         data_tr = avi_base('%sSub%s_1_multitarget.mat' % (path, subject))
43         data_te = avi_base('%sSub%s_2_multitarget.mat' % (path, subject))
44     else:
45         data_te = avi_base('%sSub%s_1_multitarget.mat' % (path, subject))
46         data_tr = avi_base('%sSub%s_2_multitarget.mat' % (path, subject))
47     features_tr, features_te = feature_extraction(data_tr, data_te)
48     # aplicando o pre-processamento usando uniformização dos dados nas
49     # características extraídas.
50     ss = StandardScaler()
51     for model in features_tr:
52         model.data = ss.fit_transform(model.data)
53     for model in features_te:
54         model.data = ss.transform(model.data)
55     # treinando o classificador SVM
56     estimator = svm.SVC(kernel='linear', C=2) # utilizando SVM como estimador
57     for model in features_tr:
58         model.training(estimator)
59     # teste utilizando a 1ª sessão (menor variabilidade de classes)
60     n_t = len(features_te[0].labels) # number of tests
61     # reformatando vetor para aplicação dos testes

```

```

62     tests = np.array([[m.data[e] for m in features_te] for e in range(n_t)])
63     # aplicando o classificador e realizando o voto
64     votes = list()
65     labels = [f.target for f in features_te]
66     for test in tests:
67         vote_set = list()
68         for model in features_tr:
69             vote_set.append(model.classifier.predict(test))
70         votes.append(labels[sum(vote_set).argmax()]) # most voted index
71     # mostrando os resultados
72     template = [data_te[i].event_id.keys()[0] for i in range(len(data_te))]
73     score = 0
74     for t, v in zip(template, votes):
75         score += 1 if t == v else 0
76     print '%.1f %%' % (score * 100 / len(template))
77     print template
78     print votes
79     # visualização da densidade espectral de potência (PSD)
80     from bci_utils import view_psd_multitaper
81     for i in range(len(data_te)):
82         view_psd_multitaper(data_te[i], 3, 11)
83
84 if __name__ == '__main__':
85     # ... chamada de função igual ao 1º experimento single target

```

A.3 Código Arduino para o desenvolvimento do semáforo com frequência cintilante

```

1  short int leds[] = {3, 5, 6}; // pinos - vermelho(3) laranja(5) verde(6)
2  short int pushButton = 11; // controle início/fim feedback
3  unsigned long ctrl_rate = millis(); // controla a frequência de cada LED
4  unsigned long time_delay = millis(); // controla o tempo de cada LED
5  float intervals[3]; // intervalos para a taxa de frequência
6  short int led_value = LOW; // valor inicial do LED vazio
7  short int current_led = -1; // variável de índice inicial do LED
8  short int count = -1; // indexa a sequência e controla o fim
9  short int size_seq; // controla o tamanho da sequência
10 bool running = false; // controle do feedback
11 // ##### CONFIGURAÇÕES #####
12 // Sequência gerada aleatoriamente para acendimento dos LEDs
13 short int sequence[] = {1, 1, 0, 2, 1, 2, 2, 2, 0, 0, 1, 2, 2, 0, 1, 0, 2, 0, 2, 2};
14 short int frequencias[] = {8, 10, 12}; // frequências do 2º conj. exp.
15 // short int frequencias[] = {37, 40, 43}; // frequências da 2ª conf. do 3º conj. exp.
16 // short int frequencias[] = {40, 40, 40}; // frequências da 3ª conf. do 3º conj. exp.
17 short int time_led = 6000; // tempo de cada LED piscando (ms)
18 short int rest_time = 3000; // não faz nada durante a troca de LED (ms)

```

```
19 short int start_time = 2000;           // tempo para iniciar o experimento (ms)
20 // #####
21
22 void reset()
23 {
24     running = false; // flag que controla se o experimento está rodando
25     digitalWrite(leds[current_led], LOW);
26     current_led = -1; // primeiro incremento inicia em 0 (correto)
27     count = -1;      // primeiro incremento inicia em 0 (correto)
28     Serial.print(0); // flag que indica o fim dos marcadores
29     delay(1000);     // ignora o tempo que o botão fica ativo
30 }
31
32 void setup()
33 {
34     Serial.begin(9600); // inicia porta serial
35     for (short int i = 0; i < 3; i++)
36     {
37         pinMode(leds[i], OUTPUT); // configura os LEDs de saída
38         // cálculo do intervalo de tempo de acordo com a frequência
39         intervals[i] = ((1.0 / frequencias[i]) / 2) * 1000;
40     }
41     // configura botão que controla o experimento como entrada
42     pinMode(pushButton, INPUT);
43     digitalWrite(pushButton, HIGH);
44     // soma de ambos os tempos 'delay' ('millis' não para em 'delay')
45     time_led += rest_time;
46     // tamanho da sequência de LEDs
47     size_seq = sizeof(sequence) / sizeof(short int);
48 }
49
50 void loop()
51 {
52     start:
53     // lê o estado do botão
54     short int buttonState = digitalRead(pushButton);
55     // executa quando o 'pushButton' é pressionado
56     if (!buttonState)
57     {
58         if (!running)
59         {
60             running = true;
61             Serial.print(1);
62
63             delay(start_time);
64         }
65         else
66             reset();
67     }
68 }
```

```

66  if (running)
67  {
68      unsigned long time_now = millis();
69      // controla o tempo de evocação da frequência
70      if (current_led == -1 || (time_now - time_delay > time_led))
71      {
72          // zera todo os LED quando troca de estado
73          for (short int i = 0; i < 3; i++)
74              digitalWrite(leds[i], LOW);
75          count++;
76          if (count == size_seq)
77          {
78              reset();
79              goto start;
80          }
81          current_led = sequence[count];
82          if (count)
83              Serial.print(2);    // registra o intervalo nos marcadores
84          delay(rest_time);
85          time_delay = time_now;
86          // envia o código do LED aceso como marcador
87          Serial.print(leds[current_led]);
88      }
89      // controla o tempo de frequência de cintilação
90      if (time_now - ctrl_rate > intervals[current_led])
91      {
92          ctrl_rate = time_now;
93          if (led_value == LOW)
94              led_value = HIGH;
95          else
96              led_value = LOW;
97          digitalWrite(leds[current_led], led_value);
98      }
99  }
100 }

```

A.4 Código Arduino para encontrar a máxima frequência cintilante não visível ao olho humano

```

1  long ctrl_rate = millis();    // controla a frequência do LED
2  long time_delay = millis();  // controla o tempo do LED
3  short int pushButton = 11;   // controle início/fim do experimento
4  float interval;              // intervalo para a taxa de frequência
5  short int led_value = LOW;    // valor inicial do LED baixo
6  bool running = false;        // controla se o experimento está rodando
7  // CONFIGURAÇÕES #####

```

```
8  short int led = 6;           // pino utilizado para o LED
9  short int time_led = 5000;   // tempo em que o LED ficará piscando (ms)
10 short int rest_time = 2000;  // não faz nada durante a troca de frequência de LED (ms)
11 short int start_time = 2000; // tempo para iniciar o experimento (ms)
12 short int frequency = 29;    // valor inicial da frequência (contado +1 no início)
13
14 void reset()
15 {
16     Serial.print("Parou com a frequência de ");
17     Serial.print(frequency);
18     Serial.println(" Hz");
19     running = false;
20     digitalWrite(led, LOW);
21     // necessário para ignorar o tempo utilizado ao pressionar o botão
22     delay(1000);
23     frequency = 29;
24 }
25
26 void setup()
27 {
28     Serial.begin(9600);
29     interval = ((1.0 / frequency) / 2) * 1000;
30     pinMode(led, OUTPUT);
31     pinMode(pushButton, INPUT);
32     digitalWrite(pushButton, HIGH);
33     // soma ambos os tempos, pois o tempo de espera continua quando
34     // configurado com 'millis()'
35     time_led += rest_time;
36 }
37
38 void loop()
39 {
40     // 'Push Button' para controlar quando o início do experimento e o fim
41     // que indica quando o participante não está mais vendo as frequências de
42     // cintilação de forma nítida.
43     short int buttonState = digitalRead(pushButton);
44     if (!buttonState)
45     {
46         if (!running)
47         {
48             running = true;
49             delay(start_time);
50         }
51         else
52             reset();
53     }
54     if (running)
```

```

55  {
56      long time_now = millis();
57      // controle de tempo do LED em diferentes frequências de cintilação
58      if (frequency == 29 || (time_now - time_delay > time_led))
59      {
60          digitalWrite(led, LOW);
61          interval = ((1.0 / ++frequency) / 2) * 1000;
62          delay(rest_time);
63          Serial.print("Frequência atual: ");
64          Serial.println(frequency);
65          time_delay = time_now;
66      }
67      // controle do tempo entre as cintilações
68      if (time_now - ctrl_rate > interval)
69      {
70          ctrl_rate = time_now;
71          if (led_value == LOW)
72              led_value = HIGH;
73          else
74              led_value = LOW;
75          digitalWrite(led, led_value);
76      }
77  }
78  }

```

A.5 Código para a gravação dos dados de EEG e marcadores no 2º conjunto experimental

```

1  from commands import getoutput as gop
2  from datetime import datetime
3  import json
4  import multiprocessing as mp
5  import socket
6  import serial
7  import mne
8  import numpy as np
9
10 # "3": LED vermelho ligado
11 # "5": LED laranja ligado
12 # "6": LED verde ligado
13 LABELS = {3: 1, 5: 2, 6: 3}
14
15 def get_markers(queue):
16     ''' função para conectar na porta serial para receber os marcadores. '''
17
18     device = gop('ls /dev/ | grep ACM')

```

```

19     print device
20     conn = serial.Serial('/dev/%s' % device, 9600)
21     print 'Ready to receive!'
22     marker = -1
23     while marker:
24         queue.put(int(conn.read()))
25
26 def make_fif_file(data):
27     ''' função para criar o arquivo MNE FIF. '''
28
29     # criando os dados 'info'
30     info = mne.create_info(
31         ch_names=['O1', 'Oz', 'O2', 'P07', 'P03', 'P04', 'P08', 'Pz'],
32         ch_types=['eeg'] * 8,
33         sfreq=250,
34         montage='standard_1020'
35     )
36     # criando os eventos (rótulos)
37     events = list()
38     for i, m in enumerate(data[0]):
39         events.append([i, 0, m])
40     events = np.array(events)
41     event_id = dict(red=1, yellow=2, green=3)
42     # corte dos dados excedentes da sessão para o tamanho mínimo
43     minor = min([len(item) for item in data[1]]) # obtém o mínimo
44     # construindo o formato apropriado
45     data = np.array([d[:minor] for d in data[1]])
46     # configurando 'MNE epochs' (n_epochs, n_chans, n_times)
47     data = data.swapaxes(1, 2)
48     # criando os objetos 'epochs'
49     epochs = mne.EpochsArray(data=data, info=info, events=events,
50                             event_id=event_id)
51     # criar arquivo FIF com o nome do experimento/participante
52     dt = datetime.now()
53     dt = dt.strftime('%d-%m-%y_%H%Mmin%Ss')
54     user = raw_input('Participante: ')
55     n_exp = raw_input('Número do experimento: ')
56     exp = 'SSVEP-Exp{}'.format(n_exp)
57     epochs.save('{}_{}-epo.fif'.format(exp, user, dt))
58
59 def exit_all(proc, client):
60     '''Finaliza as conexões e sai do programa.'''
61
62     print '\tKeyboard interrupt received'
63     proc.terminate()
64     client.close()
65     exit()

```

```
66
67 def udp_client():
68     '''Cria uma conexão UDP e retorna um objeto cliente.'''
69
70     ip = "127.0.0.1"
71     port = 12345
72     client = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
73     client.bind((ip, port))
74     return client
75
76 def build_data():
77
78     # cria uma fila compartilhada para recebimento dos marcadores
79     queue = mp.Queue()
80     # inicia um subprocesso para recebimento dos marcadores
81     proc = mp.Process(target=get_markers, args=(queue,))
82     proc.daemon = True
83     proc.start()
84     client = udp_client()
85     # ponto de espera até receber uma flag inicial '1'
86     try:
87         queue.get()
88     except KeyboardInterrupt:
89         exit_all(proc, client)
90     data = [[], []]
91     trial = list()
92     print 'Recebendo dados EEG e marcadores...'
93     while True:
94         try:
95             # recebe e decodifica o código JSON
96             json_data, _ = client.recvfrom(1024)
97             sample = json.loads(json_data)
98             trial.append(sample['data'])
99             if queue.qsize():
100                 marker = queue.get()
101                 # somente se marcador final '0' ou intervalo de descanso '2'
102                 # Marker end "0" or rest interval "2"
103                 if marker in (0, 2):
104                     data[1].append(trial)
105                     if not marker:
106                         break
107                 else:
108                     trial = list()
109                     data[0].append(LABELS[marker])
110         except KeyboardInterrupt:
111             exit_all(proc, client)
112     make_fif_file(data)
```



```
113
114 if __name__ == '__main__':
115     build_data()
```

A.6 Código para a validação e análise dos resultados do 2º conjunto experimental

```
1 from mne.decoding import PSDEstimator
2 from mne.filter import filter_data, notch_filter
3 from sklearn.feature_selection import RFE
4 from sklearn.preprocessing import StandardScaler
5 from sklearn import svm
6 from mne import set_eeg_reference as car
7 import mne
8 import numpy as np
9 import sys
10
11 reload(sys)
12 sys.setdefaultencoding('utf8')
13
14 class Model:
15     # ... classe igual ao 1º experimento single target
16
17 def feature_extraction(data, inst):
18     '''Função desenvolvida para realizar a extração de características. Retorna
19 um vetor de características.'''
20
21     # configurando o desvio padrão
22     sd = 0.3
23     sfreq = inst.info['sfreq']
24     features = list()
25     for target in inst.event_id:
26         y = list()
27         for i in range(len(inst)):
28             y.append(int(target == inst[i].event_id.keys()[0]))
29             target_value = {'red': 8.0, 'yellow': 10.0, 'green': 12.0}
30             fmin = float(target_value[target]) - sd
31             fmax = float(target_value[target]) + sd
32             psd = PSDEstimator(sfreq=sfreq, fmin=fmin, fmax=fmax)
33             X = psd.transform(data)
34             features.append(Model(X, target, y))
35     return features
36
37 def filters(data, fs, length):
38     '''Aplicação de filtros temporais.'''
39
```

```
40     # configurando filtro "notch" nas frequências de 60 e 120 Hz
41     notches = [60, 120]
42     # configurando filtro passa-faixas entre 5 e 50 Hz
43     low, high = 5., 50.
44     # aplicação do filtro 'notch'
45     filtered = notch_filter(x=data, Fs=fs, freqs=notches,
46                             filter_length='%ss' % length)
47     # aplicação do filtro 'passa faixas'
48     filtered = filter_data(filtered, fs, low, high)
49     return filtered
50
51 def main(data_ori, **ids):
52     data, inst, feature = dict(), dict(), dict()
53     # separando os dados em treino e teste
54     data['tr'] = deepcopy(data_ori).drop(ids['te'])
55     data['te'] = deepcopy(data_ori).drop(ids['tr'])
56     for tp in ('tr', 'te'):
57         # aplicando a Referência média comum (CAR)
58         inst[tp], data[tp] = car(data[tp],
59                                 ref_channels=['Oz', 'O2', 'P04', 'P07'])
60         # aplicando os filtros temporais
61         data[tp] = filters(data[tp], inst[tp].info['sfreq'], inst[tp].tmax)
62         # extraíndo as características
63         feature[tp] = feature_extraction(data[tp], inst[tp])
64     # aplicando o 'StandardScaler'
65     ss = StandardScaler()
66     for model in feature['tr']:
67         model.data = ss.fit_transform(model.data)
68     for model in feature['te']:
69         model.data = ss.transform(model.data)
70     # aplicando o treinamento do SVM e realizando o voto
71     estimator = svm.SVC(kernel='linear', C=2)
72     for model in feature['tr']:
73         model.training(estimator)
74     n_t = len(feature['te'][0].labels)
75     tests = np.array([m.data[e] for m in feature['te']] for e in range(n_t)])
76     votes = list()
77     labels = [f.target for f in feature['te']]
78     for test in tests:
79         vote_set = list()
80         for model in feature['tr']:
81             vote_set.append(model.classifier.predict(test))
82         votes.append(labels[sum(vote_set).argmax()])
83     temp = [inst['te'][i].event_id.keys()[0] for i in range(len(inst['te']))]
84     score = 0
85     for t, v in zip(temp, votes):
86         score += 1 if t == v else 0
```

```

87     score = '%.1f %%' % (score * 100 / len(temp))
88     print temp
89     print votes
90     return score
91
92 if __name__ == '__main__':
93     from sys import argv
94     if len(argv) > 1:
95         data = mne.read_epochs(argv[1])
96         pc_tr = int(len(data) * 0.3)
97         scores = list()
98         for i, e in enumerate(range(pc_tr, len(data) + 1)):
99             ids_tr = list(set(range(i, e)))
100             ids_te = list(set(range(len(data))).difference(ids_tr))
101             scores.append(main(data, tr=ids_tr, te=ids_te))
102     print scores

```

A.7 Código para a validação e análise dos resultados do 3º conjunto experimental

```

1 from mne.decoding import PSDEstimator
2 from mne.filter import filter_data, notch_filter
3 from sklearn.model_selection import KFold, ParameterGrid
4 from sklearn.feature_selection import RFE
5 from sklearn.preprocessing import StandardScaler
6 from sklearn import svm
7 from mne import set_eeg_reference as car
8 import mne
9 import numpy as np
10 import sys
11 # função desenvolvida para apresentação dos gráficos
12 from bci_utils import view_psd_multitaper
13 from bci_utils import view_psd
14
15 reload(sys)
16 sys.setdefaultencoding('utf8')
17
18
19 class Model:
20     # ... classe igual ao 1º experimento single target
21
22     def feature_extraction(data, inst):
23         '''Função desenvolvida para realizar a extração de características. Retorna
24         um vetor de características.'''
25
26         # configurando o desvio padrão

```

```

27     sd = 1.0
28     sfreq = inst.info['sfreq']
29     features = list()
30     for target in inst.event_id:
31         y = list()
32         for i in range(len(inst)):
33             y.append(int(target == inst[i].event_id.keys()[0]))
34             target_value = {'red': 8.0, 'yellow': 10.0, 'green': 12.0}
35             # target_value = {'red': 37.0, 'yellow': 40.0, 'green': 43.0}
36             fmin = float(target_value[target]) - sd
37             fmax = float(target_value[target]) + sd
38             psd = PSDEstimator(sfreq=sfreq, fmin=fmin, fmax=fmax)
39             X = psd.transform(data)
40             features.append(Model(X, target, y))
41     return features
42
43 def filters(data, fs, length, tb):
44     '''Aplicação de filtros temporais.'''
45
46     # configurando filtro "notch" nas frequências de 60 e 120 Hz
47     notches = np.arange(60, 121, 60)
48     # configurando filtro passa-faixas entre 5 e 50 Hz
49     low, high = 5., 50.
50     # aplicando filtro notch
51     filtered = notch_filter(x=data, Fs=fs, freqs=notches, verbose=False,
52                             filter_length='%ss' % length, trans_bandwidth=tb)
53     # aplicando filtro passa-faixas
54     filtered = filter_data(filtered, fs, low, high, verbose=False,
55                             filter_length='%ss' % length)
56     return filtered
57
58 def svm_classifier(feature, inst, C):
59     '''Treinamento, classificação e resultado.'''
60
61     # aplicando o treinamento do SVM
62     estimator = svm.SVC(kernel='linear', C=C)
63     for model in feature['tr']:
64         model.training(estimator)
65     # obtendo o número de testes
66     n_t = len(feature['te'][0].labels)
67     # refatorando os dados e aplicando o teste
68     tests = np.array([[m.data[e] for m in feature['te']] for e in range(n_t)])
69     # aplicando o classificador e realizando o voto
70     votes = list()
71     labels = [f.target for f in feature['te']]
72     for test in tests:
73         vote_set = list()

```

```

74         for model in feature['tr']:
75             vote_set.append(model.classifier.predict(test))
76             # recuperando índices mais votados
77             votes.append(labels[sum(vote_set).argmax()])
78         temp = [inst['te'][i].event_id.keys()[0] for i in range(len(inst['te']))]
79         score = 0
80         for t, v in zip(temp, votes):
81             score += 1 if t == v else 0
82         res = score * 100 / len(temp)
83         return res
84
85 def svm_optimize_param(feature, inst):
86     '''Otimização do parâmetro "C" da SVM.'''
87
88     best_result = 0
89     selected_C = None
90     # iteração sobre uma faixa de valores para o parâmetro "C"
91     for C in np.arange(.5, 10.1, .5):
92         res = svm_classifier(feature, inst, C)
93         if res >= best_result:
94             best_result = res
95             selected_C = C
96     return selected_C
97
98 def main(fname1, fname2, tmax):
99     data_epo1 = mne.read_epochs(fname1, verbose=False)
100    data_epo2 = mne.read_epochs(fname2, verbose=False)
101    # obtém o menor "tempo máximo" das duas sessões
102    min_time_epo = min(data_epo1.tmax, data_epo2.tmax)
103    # padroniza as sessões para o menor tempo encontrado e as concatena
104    data_epo = mne.concatenate_epochs((data_epo1.crop(0, min_time_epo),
105                                       data_epo2.crop(0, min_time_epo)))
106    # Relacionando o tempo (em segundos) com as bandas de transição (em Hz)
107    # para serem aplicadas aos filtros
108    tmax_tb = {6.0: 1.2, 5.0: 1.4, 4.0: 1.7, 3.0: 2.2, 2.0: 3.3}
109    tb = tmax_tb[tmax]
110    if data_epo.tmax < tmax:
111        tmax = data_epo.tmax
112    data_epo.crop(0, tmax)
113    # contrói um gerador com 8 folds para realizar uma validação cruzada
114    kf = KFold(n_splits=8)
115    # iterações dos K folds na validação cruzada
116    i_fold = 0
117    results = list()
118    for train_i, test_i in kf.split(data_epo):
119        data, inst, feature = dict(), dict(), dict()
120        data['tr'], data['te'] = data_epo[train_i], data_epo[test_i]

```

```

121     for tp in ('tr', 'te'):
122         # aplicando a Referência média comum (CAR)
123         inst[tp], data[tp] = car(data[tp], verbose=False,
124                                 ref_channels=ch)
125         # aplicando os filtros temporais
126         data[tp] = filters(data[tp], inst[tp].info['sfreq'], tmax, tb)
127         # extraíndo as características
128         feature[tp] = feature_extraction(data[tp], inst[tp])
129         # otimizando parâmetros da SVM com 80% dos dados de treinamento
130         n_ex_param = int(len(inst['tr']) * .8)
131         feature_param = {'tr': list(), 'te': list()}
132         ss = StandardScaler()
133         # criando o modelo de treinamento para o conjunto de dados utilizado
134         # para a otimização do parâmetro da SVM, aplicando o Standard Scaler
135         for t, f in zip(inst['tr'].event_id, feature['tr']):
136             X = ss.fit_transform(f.data[:n_ex_param])
137             feature_param['tr'].append(Model(X, t, f.labels[:n_ex_param]))
138             y = ss.transform(f.data[n_ex_param:])
139             feature_param['te'].append(Model(y, t, f.labels[n_ex_param:]))
140         inst_param = dict()
141         inst_param['tr'] = inst['tr'][:n_ex_param]
142         inst_param['te'] = inst['tr'][n_ex_param:]
143         # obtendo o parâmetro C otimizado
144         C = svm_optimize_param(feature_param, inst_param)
145         # aplicando o 'StandardScaler' para o fold completo
146         ss = StandardScaler()
147         for model in feature['tr']:
148             model.data = ss.fit_transform(model.data)
149         for model in feature['te']:
150             model.data = ss.transform(model.data)
151         # treinando, classificando e obtendo resultado
152         res = svm_classifier(feature, inst, C)
153         results.append(res)
154         i_fold += 1
155         print 'Resultado (fold %d) (C = %f): %.1f %%' % (i_fold, C, res)
156         # mostrando a média dos resultados
157         print 'Média: %.1f %%' % (sum(results) / i_fold)
158
159 if __name__ == '__main__':
160     from sys import argv
161     if len(argv) > 3:
162         # carrega os dois arquivos de entrada
163         main(argv[1], argv[2], float(argv[3]))
164     elif len(argv) == 1:
165         print '\nNecessário o 1º e 2º arquivo FIF e tempo máximo por parâmetro.'
166     elif len(argv) == 2:
167         print '\nNecessário o 2º arquivo FIF e tempo máximo por parâmetro.'

```

```

168     elif len(argv) == 3:
169         print '\nNecessário acrescentar o tempo máximo por parâmetro.'

```

A.8 Biblioteca utilizada para geração dos gráficos de PSD

```

1 from mne.time_frequency import psd_multitaper
2 import numpy as np
3 import matplotlib.pyplot as plt
4
5 def view_psd(epoch, fmin, fmax, tmin, tmax):
6     ax = plt.axes()
7     values = {1: 8, 2: 10, 3: 12}
8     # values = {1: 37, 2: 40, 3: 43}
9     evoked_value = values[epoch.events[0][2]]
10    ax.axvline(evoked_value, linewidth=2, c='r', ls='dashed')
11    ax.set_title('Evocado %s Hz' % str(evoked_value), fontsize=16,
12                fontweight='bold')
13    ax.set_ylabel('Densidade Espectral de Potência - uV/Hz (dB)'.decode('utf-8'))
14    ax.set_xlabel('Frequência (Hz)'.decode('utf-8'))
15    # print help(epoch.plot_psd)
16    epoch.plot_psd(fmin=fmin, fmax=fmax, tmin=tmin, tmax=tmax, color=(0, 0, 1),
17                  n_jobs=8, ax=ax, area_mode='range')
18
19 def view_psd_multitaper(epoch, fmin, fmax, tmin, tmax):
20     psds, freqs = psd_multitaper(epoch, fmin=fmin, fmax=fmax, tmin=tmin,
21                                  tmax=tmax)
22     ax = plt.axes()
23     values = {1: 8, 2: 10, 3: 12}
24     # values = {1: 37, 2: 40, 3: 43}
25     evoked_value = values[epoch.events[0][2]]
26     ax.axvline(evoked_value, linewidth=2, c='#ff7f00', ls='dashed')
27     ax.plot(freqs, psds[0][0], color='#2a76cc')
28     ax.set(title='Evocado %s Hz' % str(evoked_value),
29            ylabel='Densidade Espectral de Potência (Multitaper)'.decode('utf-8'),
30            xlabel='Frequência (Hz)'.decode('utf-8'))
31     plt.show()

```

A.9 Código para estimar o valor médio de PSD do sinal SSVEP evocado por cores de LED

```

1 from mne.decoding import PSDEstimator
2 from mne.filter import filter_data, notch_filter
3 import mne
4 import numpy as np
5 import statistics as stats

```

```
6 import sys
7 # função desenvolvida para apresentação dos gráficos
8 from bci_utils import view_psd_multitaper
9 from bci_utils import view_psd
10
11 reload(sys)
12 sys.setdefaultencoding('utf8')
13
14 def filters(data, fs, length):
15     # ... igual as avaliações passadas
16
17 def main(fname):
18     # carrega o arquivo FIF
19     data_epo = mne.read_epochs(fname, verbose=False)
20     # remove canais não utilizados (sobra somente 'Oz')
21     dropped_chs = ['O1', 'O2', 'P07', 'P03', 'P04', 'P08', 'Pz']
22     data_epo.drop_channels(dropped_chs)
23     # obtendo a taxa de frequência
24     sfreq = data_epo.info['sfreq']
25     fmin, fmax = 39, 41
26     psd = PSDEstimator(sfreq=sfreq, fmin=fmin, fmax=fmax)
27     data, X = dict(), dict()
28     for event in ('green', 'red', 'yellow'):
29         # filtrando os dados separado em eventos
30         data[event] = filters(data_epo[event].get_data(),
31                               data_epo.info['sfreq'], data_epo.tmax)
32         # extraíndo os psds separado em eventos
33         X[event] = psd.transform(data[event])
34         print event,
35         res = list()
36         for trial in X[event]:
37             res.append(stats.mean(trial[0]))
38         # mostra o calculo da média e desvio padrão
39         print stats.mean(res), stats.stdev(res)
40         res = list()
41
42 if __name__ == '__main__':
43     from sys import argv
44     if len(argv) > 1:
45         main(argv[1])
46     elif len(argv) == 0:
47         print 'Necessário passar arquivo FIF por parâmetro.'
```


APÊNDICE B – Termo de Consentimento Livre e Esclarecido

Termo de Consentimento Livre e Esclarecido

Título da pesquisa: Desenvolvimento de um sistema baseado em uma Interface Cérebro-Computador para auxiliar na tomada de decisões

Pesquisador responsável pela pesquisa, com Endereços e Telefones: Rodrigo Hübner, endereço: Rua Santo Antônio, 83, apto 203, Maringá/PR, CEP 87030-130. telefone: (44) 99931-6040.

Local de realização da pesquisa: Departamento de Informática da Universidade Estadual de Maringá, Bloco C56.

Endereço, telefone do local: Av. Colombo, 5790 – Jd. Universitário, CEP 87020-900, Maringá/PR. Telefone: (44) 3011-4324.

A) INFORMAÇÕES AO PARTICIPANTE

1. Apresentação da pesquisa.

Neste experimento iremos investigar novas estratégias aplicadas no estudo de Interfaces Cérebro-Computador (ICC), utilizando estímulos visuais. Vamos trabalhar com um paradigma da ICC chamado SSVEP (Potenciais Evocados em Regime Estacionário). Iremos utilizar um protótipo de semáforo com *Light Emitting Diode* (LED) que constitui um sistema para auxiliar na tomada de decisões, de modo que o alvo não se mostre piscando ao participante. Desta forma, na primeira parte da experimentação iremos aplicar um sistema SSVEP, no qual os LEDs ficarão piscando em uma frequência baixa. Após isso, na segunda parte experimental, serão configuradas frequências não visíveis e ausência de frequência com a luzes do protótipo de semáforo acesas constantemente. Registraremos os sinais de eletroencefalograma (EEG) durante todo o momento da experimentação. Toda a experimentação será realizada no Departamento de Informática da Universidade Estadual de Maringá, Bloco C56, no qual você será informado previamente o dia, horário e sala que o experimento será realizado.

2. Objetivos da pesquisa.

O objetivo no qual você está convidado é avaliar novas estratégias aplicadas ao paradigma SSVEP apresentado em sistemas de Interface Cérebro-Computador (ICC), almejando o desenvolvimento de um sistema SSVEP-ICC para auxiliar na tomada de decisões em situações próximas do mundo real, no qual um protótipo de semáforos construído com LEDs, será utilizado para criar tais situações.

3. Participação na pesquisa.

Para que os os objetivos desta pesquisa sejam cumpridos, você será submetido aos experimentos em dois dias diferentes. Em todos os experimentos, você ficará sentado em uma cadeira em frente a um protótipo de semáforo de trânsito, posicionado em cima de uma mesa a 60 cm de seus olhos. Além disso, utilizará um capacete com oito eletrodos posicionados levemente em seu couro cabeludo e duas presilhas de eletrodos presa uma em cada lobo da orelha. O capacete possui um tamanho único, porém os eletrodos são ajustáveis a ponto se se adaptar à superfície de sua cabeça. Os eletrodos dispensam a utilização de pasta ou gel eletrolítico, permitindo assim a fácil colocação

em você durante o experimento e eles **não são descartáveis** e são higienizados antes do uso por outra pessoa. O protótipo possui três diferentes LEDs alto-brilho de 5 mm, nas cores verde, vermelho e laranja, no qual terá que focar nas luzes que estiverem acesas durante a sessão. As luzes que acenderão não ficarão direcionadas diretamente para os seus olhos (para não causar desconforto visual). No primeiro dia, você realizará dois diferentes experimentos. Cada um dos experimentos contemplará duas sessões com configurações iguais, no qual cada sessão terá a duração de quatro minutos. Em cada sessão acenderá aleatoriamente uma das três luzes vinte vezes por oito segundos, com intervalo de quatro segundos entre cada luz acesa. De uma sessão para outra terá um tempo de descanso de aproximadamente dois minutos ou mais, ficando ao seu critério.

Após o final dos dois primeiros experimentos no primeiro dia, você será convidado a olhar para uma das luzes do semáforo (agora sem o equipamento de EEG) e responder em qual momento não é possível visualizar a luz piscando. Uma das luzes começará a piscar vagarosamente e irá aumentando a frequência em que pisca a cada 3 segundos. Esta informação será utilizada para configurar a experimentação do segundo dia. Damos a este experimento o nome de Enquete da Frequência Não-Visível (EFNV).

No segundo dia, você realizará mais dois experimentos com duas sessões cada. Tais experimentos terão as mesmas durações que os realizados no primeiro dia. O terceiro experimento será realizado com cada LED configurado com a frequência média em resposta a EFNV obtida ao final do primeiro dia de experimentação. Já no quarto experimento, cada LED será configurado com diferentes frequências em torno da EFNV analisada.

Ao final de cada dia de experimento, será perguntado a você o nível da fadiga visual (o nível de desconforto visual) na escala de 0 à 10.

4. **Confidencialidade.**

Os pesquisadores responsáveis por este estudo garantem o sigilo e a privacidade de todo e qualquer dado coletado nesta pesquisa, podendo esses dados serem utilizados para a produção de artigos científicos. Mesmo nessa hipótese, o nome completo, a imagem ou qualquer outra informação que possa identificá-lo não será publicada sem sua prévia autorização.

5. **Riscos e Benefícios.**

5a) Riscos: O equipamento que realiza a interface cérebro-computador utilizando um eletroencefalograma, dispõe de um procedimento não invasivo, o que não gera maiores riscos. Será utilizado um capacete, no qual cada eletrodo se ajusta com a área do couro cabeludo do participante, o que pode causar algum desconforto depois de utilizá-lo por muito tempo. A necessidade de haver o contato do eletrodo com o couro cabeludo, também pode ser um desconforto se o participante tiver uma quantidade densa de cabelo, no qual precisará ser afastado para manter o contato.

Outro desconforto que você poderá ter durante o experimento, é com as luzes de LED piscando no protótipo. Este desconforto será reduzido, configurando o ambiente de experimentação para que os LEDs não fiquem apontados diretamente para o seu olho, sugerindo intervalos de descanso suficientes entre uma sessão e outra a critério do participante. Tais precauções não influenciarão no resultado dos experimentos.

5b) Benefícios: Nesta pesquisa queremos entender como se comportam os indivíduos em relação aos sinais cerebrais obtidos de um eletroencefalograma, quando são

submetidos em situação de tomada de decisão. Para isso foi criado um protótipo de semáforo, no qual é utilizado como estímulo visual. O benefício empregado nesta pesquisa é de criar um sistema baseado na Interface Cérebro-Computador que possa classificar os diferentes estímulos apresentados (para o exemplo aplicado nesta pesquisa, as luzes vermelha, amarela e verde do semáforo) e auxiliar na tomada de decisão (neste caso, parar, reduzir ou seguir em frente com o veículo). Esta pesquisa também poderá ser aplicada em outras situações no qual o estímulo ocorre visualmente, por exemplo, a lanterna traseira sinalizando a frenagem de um carro à frente. Em outras palavras, o principal benefício desta pesquisa é auxiliar o utilizador do sistema a ser desenvolvido em situações potencialmente de risco, sendo assim, **um benefício a ciência e não um benefício individual**.

6. Critérios de inclusão e exclusão.

6a) Inclusão: Homens e mulheres com idade acima de dezoito anos e saudáveis estão aptos à realização da pesquisa.

6b) Exclusão: Estarão excluídos da participação do experimento, sujeitos que possuírem diagnósticos de baixa visão ou cegueira (por se tratar de um experimento com estímulos visuais), sujeitos diagnosticados com qualquer transtorno psiquiátrico (ansiedade, depressão, esquizofrenia, hiperatividade, epilepsia, etc) devido à alterações de condução sináptica nesses pacientes. O experimento é contraindicado a pessoas que tenham seborreia excessiva, infecções de pele no couro cabeludo ou pediculose.

7. Direito de sair da pesquisa e a esclarecimentos durante o processo.

Informamos que você possui a liberdade de se recusar a participar do estudo ou de retirar seu consentimento a qualquer momento, sem precisar qualquer justificativa e sem que você sofra qualquer prejuízo. Da mesma maneira você poderá retirar seu consentimento para a participação na pesquisa a qualquer momento, sem nenhum prejuízo. Além disso, é assegurada a sua assistência durante toda a pesquisa, bem como o livre acesso a todas as informações e esclarecimentos adicionais sobre o estudo e suas consequências.

Você pode assinalar o campo a seguir, para receber o resultado desta pesquisa, caso seja de seu interesse:

- () quero receber os resultados da pesquisa (e-mail para envio: _____)
() não quero receber os resultados da pesquisa

8. Ressarcimento e indenização.

Não haverá qualquer despesa decorrente da sua participação na pesquisa e, da mesma forma, você não receberá qualquer valor de ressarcimento por sua participação. Ainda assim, caso ocorra algum dano decorrente de sua participação na pesquisa, você será devidamente indenizado, conforme determina a lei.

ESCLARECIMENTOS SOBRE O COMITÊ DE ÉTICA EM PESQUISA:

O Comitê de Ética em Pesquisa envolvendo Seres Humanos (CEP) é constituído por uma equipe de profissionais com formação multidisciplinar que está trabalhando para assegurar o respeito aos seus direitos como participante de pesquisa. Ele tem por

objetivo avaliar se a pesquisa foi planejada e se será executada de forma ética. Se você considerar que a pesquisa não está sendo realizada da forma como você foi informado ou que você está sendo prejudicado de alguma forma, entre em contato com o Comitê de Ética em Pesquisa envolvendo Seres Humanos da Universidade Tecnológica Federal do Paraná (CEP/UTFPR). **Endereço:** Av. Sete de Setembro, 3165, Bloco N, Térreo, Bairro Rebouças, CEP 80230-901, Curitiba-PR, **Telefone:** (41) 3310-4494, **e-mail:** coep@utfpr.edu.br.

B) CONSENTIMENTO

Eu declaro ter conhecimento das informações contidas neste documento e ter recebido respostas claras às minhas questões a propósito da minha participação direta (ou indireta) na pesquisa e, adicionalmente, declaro ter compreendido o objetivo, a natureza, os riscos, benefícios, ressarcimento e indenização relacionados a este estudo.

Após reflexão e um tempo razoável, eu decidi, livre e voluntariamente, participar deste estudo. Estou consciente que posso deixar o projeto a qualquer momento, sem nenhum prejuízo.

Nome Completo: _____
RG: _____ Data de Nascimento: ____/____/____ Telefone: _____
Endereço: _____
CEP: _____ Cidade: _____ Estado: _____
Assinatura: _____ Data: ____/____/____

Eu declaro ter apresentado o estudo, explicado seus objetivos, natureza, riscos e benefícios e ter respondido da melhor forma possível às questões formuladas.

Nome completo: _____
Assinatura pesquisador (a): _____ Data: ____/____/____
(ou seu representante)

Para todas as questões relativas ao estudo ou para se retirar do mesmo, poderão se comunicar com _____, via e-mail: _____ ou telefone: _____.

Contato do Comitê de Ética em Pesquisa que envolve seres humanos para denúncia, recurso ou reclamações do participante pesquisado: Comitê de Ética em Pesquisa que envolve seres humanos da Universidade Tecnológica Federal do Paraná (CEP/UTFPR)

Endereço: Av. Sete de Setembro, 3165, Bloco N, Térreo, Rebouças, CEP 80230-901, Curitiba-PR, **Telefone:** 3310-4494, **E-mail:** coep@utfpr.edu.br

APÊNDICE C – Parecer do Comitê de Ética

UNIVERSIDADE
TECNOLÓGICA FEDERAL DO**PARECER CONSUBSTANCIADO DO CEP****DADOS DO PROJETO DE PESQUISA**

Título da Pesquisa: Desenvolvimento de um sistema baseado em uma Interface Cérebro-Computador para auxiliar na tomada de decisões.

Pesquisador: Rodrigo Hübner

Área Temática:

Versão: 3

CAAE: 92010318.3.1001.5547

Instituição Proponente: UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANA

Patrocinador Principal: Financiamento Próprio
UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANA

DADOS DO PARECER

Número do Parecer: 2.867.988

Apresentação do Projeto:

Segundo os autores: “DESENHO: Será uma análise experimental nos quais 6 (seis) participantes serão submetidos a um experimento utilizando um equipamento de eletroencefalografia (EEG) para a realização de uma Interface Cérebro-Computador (Brain-Computer Interface - BCI). Cada um dos participantes será submetido aos experimentos em dois dias diferentes. Em todos os experimentos, os participantes ficarão sentados em uma cadeira em frente a um protótipo de semáforo de trânsito, posicionado em cima de uma mesa a 60 cm de seus olhos. O protótipo possui três diferentes LEDs alto-brilho de 5 mm, nas cores verde, vermelho e laranja, no qual terão que focar nas luzes que estiverem acesas durante a sessão. As luzes que acenderão não ficarão direcionadas diretamente para os olhos dos participantes para não causar desconforto visual. No primeiro dia, cada participante realizará dois diferentes experimentos. Cada um dos experimentos contemplará duas seções com configurações iguais, no qual cada sessão terá a duração de quatro minutos. Em cada seção acenderá aleatoriamente uma das três luzes vinte vezes por oito segundos, com intervalo de quatro segundos entre cada luz acesa. De uma seção para outra terá um tempo de descanso de aproximadamente dois minutos ou mais, dependendo do participante. Após o final dos dois primeiros experimentos no primeiro dia, cada participante será convidado a olhar para uma das luzes do semáforo (agora sem o equipamento de EEG) e responder em qual frequência não é possível visualizar a luz piscando. Uma das luzes começará a

Endereço: SETE DE SETEMBRO 3165

Bairro: CENTRO

UF: PR

Município: CURITIBA

Telefone: (41)3310-4494

CEP: 80.230-901

E-mail: coep@utfpr.edu.br

UNIVERSIDADE
TECNOLÓGICA FEDERAL DO

Continuação do Parecer: 2.867.988

piscar em 30 Hz e a cada cinco segundos aumenta sua frequência em uma unidade. Esta informação será utilizada para configurar a experimentação do segundo dia. Damos a este experimento o nome de Enquete da Frequência Não-Visível (EFNV). No segundo dia, cada participante realizará mais dois experimentos com duas sessões cada. Tais experimentos terão as mesmas durações que os realizados no primeiro dia. O terceiro experimento será realizado com cada LED configurado com a frequência média em resposta a EFNV obtida ao final do primeiro dia de experimentação. Já no quarto experimento, cada LED será configurado com diferentes frequências em torno da EFNV analisada.

RESUMO: Nos últimos anos, Interfaces Cérebro-Computador (ICC) passaram a ter um grande foco em sistemas fora do escopo clínico. Sistemas ICC passaram a ser utilizados para controlar equipamentos elétricos e eletrônicos, jogos digitais e outras formas de controle. Este controle pode ser realizado por meio da tomada de decisões por um sistema ICC. Um paradigma conhecido para este fim é o SSVEP, no qual é possível diferenciar alvos com diferentes frequências de cintilação, por meio de evocações visuais. Neste projeto será realizado um estudo bibliográfico de sistemas SSVEP-ICC, além de experimentos realizados com uma base de dados pública e com uma base de dados gerada por meio de um protótipo utilizando semáforos de trânsito. Os experimentos com este protótipo, permitirão criar situações de tomada de decisão, de modo que o sistema SSVEP-ICC auxilie o indivíduo a tomar decisões corretas. O objetivo deste estudo experimental é de investigar novas estratégias no modelo SSVEP -ICC para desenvolver um sistema para auxiliar na tomada de decisões aplicadas em semáforos de trânsito, de modo que o alvo não possua uma frequência de cintilação visível ao olho humano. Assim, almejamos um sistema SSVEP-ICC que seja aplicado em simulações próximas da realidade.

INTRODUÇÃO: Uma Interface Cérebro-Computador (ICC) é comumente utilizada para o desenvolvimento de sistemas que podem melhorar a qualidade de vida das pessoas que possuem algum tipo de limitação física, seja visual, auditiva ou motora. Para que isso seja possível, um sistema BCI deverá minimizar a deficiência do indivíduo auxiliando na tarefa que ele poderia realizar sozinho. Um exemplo disto são os soletradores ICC, um sistema no qual um indivíduo que possui deficiência na fala, se concentra em uma matriz de letras em um monitor, e por meio dos estímulos visuais gerados, o sistema ICC pode classificar qual letra o indivíduo está focado e exibí-la. Um sistema ICC também pode auxiliar na tomada de decisões de indivíduos saudáveis. Existem situações que podem ser consideradas de risco, por exemplo, frear um veículo enquanto está dirigindo ao ver um semáforo vermelho ou uma lanterna de freio do carro a frente acender. Em tais situações, um sistema ICC pode auxiliar o condutor, caso a decisão tomada por ele não seja a correta. Com esta premissa, a proposta deste trabalho é desenvolver um sistema ICC com o

Endereço: SETE DE SETEMBRO 3165**Bairro:** CENTRO**CEP:** 80.230-901**UF:** PR**Município:** CURITIBA**Telefone:** (41)3310-4494**E-mail:** coep@utfpr.edu.br

UNIVERSIDADE
TECNOLÓGICA FEDERAL DO

Continuação do Parecer: 2.867.988

uso do paradigma SSVEP (Steady-State Visually Evoked Potential), para determinar em qual objeto alvo um indivíduo está focado, no qual este objeto possui uma frequência de cintilação, que pode ser reconhecida com o equipamento de eletroencefalografia (EEG). Para que o sistema ICC auxilie na decisão correta, é necessário que os diferentes eventos sejam apresentados – tradicionalmente – em diferentes frequências de cintilação. Para o desenvolvimento deste trabalho, foi construído um protótipo que permite gerar o estímulo interpretado pelo SSVEP, pois quando o conceito de tomada de decisões é levado para o mundo real, tais situações não podem ser reproduzidas da mesma forma utilizando o paradigma SSVEP tradicional, pois alvos luminosos reais não apresentam uma frequência de cintilação que pode ser classificada pelo sistema ICC, além de colocar em risco a vida dos participantes do experimento. Neste contexto, o objetivo deste trabalho é apresentar um estudo empírico de técnicas utilizadas para o processamento de sinais SSVEP, almejando o desenvolvimento de um sistema para auxiliar na tomada de decisões em situações próximas do mundo real. Para cumprir com este objetivo, foi reproduzido um conjunto de experimentos baseados no paradigma SSVEP utilizando uma base de dados pública, com a intenção de avaliar os métodos de programação. Prosseguindo com os experimentos, bases de dados serão geradas por meio de aquisições de sinais EEG para serem avaliadas com um protótipo utilizando semáforos construídos com LEDs, no qual geram a evocação visual necessária para a experimentação. Por fim, será apresentado um estudo investigativo dos modelos de estimulação do sinal SSVEP para que protótipo de semáforo construído seja novamente avaliado com uma configuração mais próxima da realidade, sem a visualização de frequências de cintilação tradicionais do paradigma SSVEP. HIPÓTESE: Para este trabalho foram levantadas quatro hipóteses para a aplicação de experimentos que definem o desfecho do objetivo proposto. São elas: (1) configurar alvos (protótipo de semáforo) com diferentes frequências altas de cintilação quando estão ativos, de tal forma que estas frequências não estejam visíveis ao olho humano, (2) configurar os alvos utilizando a mesma frequência de cintilação ainda que estas frequências não sejam visíveis a olho humano, (3) analisar de forma híbrida as hipóteses 1 e 2, utilizando diferentes frequências de cintilação (como configurado em 1), somado a metodologia aplicada nos dados da configuração 1 (esta hipótese não gerará um novo experimento, pois serão utilizados os mesmos dados obtidos pela configuração 1) e, por fim (4) em que os alvos serão configurados sem frequência de cintilação, ou seja, quando as luzes estiverem ativas, elas ficarão acesas constantemente. A terceira hipótese poderá ser removida de qualquer trabalho futuro caso uma das duas primeiras hipóteses não apresente alguma relevância, pois juntas constroem a metodologia da terceira hipótese. Já a quarta hipótese também poderá ser removida de qualquer

Endereço: SETE DE SETEMBRO 3165**Bairro:** CENTRO**CEP:** 80.230-901**UF:** PR**Município:** CURITIBA**Telefone:** (41)3310-4494**E-mail:** coep@utfpr.edu.br

UNIVERSIDADE
TECNOLÓGICA FEDERAL DO

Continuação do Parecer: 2.867.988

trabalho futuro, caso o método aplicado na hipótese 2 não apresente alguma relevância. METODOLOGIA: Todos os experimentos serão realizados no Departamento de Informática da Universidade Estadual de Maringá. Os dois primeiros experimentos realizados no primeiro dia, se diferenciam entre si pelo fato do primeiro mostrar cada luz piscando em frequências baixas (entre 8 e 20 Hz) e o segundo com cada luz acesa constantemente. Ao final dos experimentos do primeiro dia, a Enquete da Frequência Não-Visível (EFNV) encontrará uma frequência alta “aparentemente” não visível aos olhos do participante. A EFNV ajudará a configurar os experimentos realizados no segundo dia. O terceiro experimento será configurado de forma que a frequência média obtida por EFNV seja aplicada igualmente aos três LEDs do alvo. Já no quarto experimento, cada LED será configurado com diferentes frequências em torno da EFNV, por exemplo, se a frequência média encontrada foi de 42 Hz, a frequência dos três LEDs serão configuradas com 42, 44 e 46 Hz. O fluxo de dados EEG é realizado por meio da ferramenta OpenBCI GUI, que transfere os dados para um algoritmo desenvolvido com a biblioteca MNE escrita em Python, que recebe os dados transferidos da ferramenta OpenBCI GUI e constrói as bases de dados utilizadas para análise futura. O equipamento de eletroencefalografia (EEG) é o OpenBCI board de 32 bits com 8 canais para a leitura de EEG mais 3 canais auxiliares utilizados para a leitura de um sensor giroscópico. O equipamento ainda pode ser expandido para 16 canais EEG utilizando módulo Daisy que acompanha o equipamento. Um capacete desenvolvido com impressora 3D (Ultracortex Mark 3), permite acoplar os eletrodos e a placa OpenBCI board. Os eletrodos utilizados são construídos com uma liga de Prata-Cloreto de Prata (Ag-AgCl), no qual dispensa a utilização de pasta ou gel eletrolítico, permitindo assim a fácil colocação do capacete em diferentes participantes durante um experimento. Vale salientar que os eletrodos não são descartáveis. Os eletrodos sofrem um desgaste com o tempo e são oxidados, porém é possível realizar todas as experimentações com os participantes, sem a necessidade de descarte dos mesmos. Já para a construção do protótipo (alvo), foi desenvolvido semáforos utilizando LEDs alto-brilho de 5 mm, sendo cada LED na cor vermelho, laranja e verde. O controle dos semáforos é realizado com o Arduino UNO, que utiliza um microcontrolador ATmega328P de 32 MB de memória flash e velocidade 16 MHz. Além dos LEDs ligados aos semáforos, foi também adicionado um botão momentâneo para controlar manualmente o início de cada sessão ou interrompê-la caso o participante julgue necessário. Para criar as frequências cintilantes, foi desenvolvido um código para o microcontrolador baseado na seguinte fórmula, no qual “f” é a frequência desejada para cada LED: $I = [1 / f] / 2 * 1000 - e$. Nesta equação é obtido o intervalo “I” de tempo entre as ativações do LED pela divisão da frequência “f” desejada por uma unidade, acrescentando a divisão

Endereço: SETE DE SETEMBRO 3165**Bairro:** CENTRO**CEP:** 80.230-901**UF:** PR**Município:** CURITIBA**Telefone:** (41)3310-4494**E-mail:** coep@utfpr.edu.br